Tektronix®

SignalVu

Vector Analysis Software

Programmer Manual

Register now!

Click the following link to protect your product.

www.tek.com/register

077-0223-14 September 2026

Copyright © 2026, Tektronix, Inc. All rights reserved. Licensed software products are owned by Tektronix or its subsidiaries or suppliers, and are protected by national copyright laws and international treaty provisions. Tektronix products are covered by U.S. and foreign patents, issued and pending. Information in this publication supersedes that in all previously published material. Specifications and price change privileges reserved.

TEKTRONIX and TEK are registered trademarks of Tektronix, Inc.

These are the original instructions in English.

Go to tek.com/eula to read the Tektronix End User License Agreement.



Contacting Tektronix

Tektronix, Inc.
13725 SW Karl Braun Drive
P.O. Box 500
Beaverton, OR 97077
USA

For product information, sales, service, and technical support:

In North America, call 1-800-833-9200.

Worldwide, visit www.tek.com to find contacts in your area.

Contents

Preface **109**
Related documentation 109

Getting Started **110**
Overview of the Manual 110
Syntax and Commands 110
Status and Events 111
Setting up remote communications 111
Measurement support 111
Remote command support 116

Syntax and Commands **118**
Command syntax 118
Backus-Naur Form Definition 118
SCPI Commands and Queries 118
IEEE 488.2 Common Commands 123
Constructed Mnemonics 123
Command groups 124
Functional groups 124
Programming hints 125
IEEE common commands 125
Obsolete commands 126
Abort commands 127
Calculate commands 128
Marker mnemonics 159
Calibration commands 160
Display commands 161
Fetch commands 196
Initiate commands 230
Input commands 230
Mass memory commands 231
Specifying the File 236
Output commands 236
Read commands 238
Sense commands 271
Status commands 316
Source commands 320
System commands 320

Trace commands . . . 322
Trace mnemonics . . . 334
Trigger commands . . . 335
Unit commands . . . 339
Retrieving response message . . . 339

Command descriptions . . . 340
ABORt (No query form) . . . 340
*CAL (Query only) . . . 340
Calculate commands . . . 341
CALCulate:[:SEARch|:TEST]:LIMit[:STATe]:MATCh:FILE:LOCation . . . 341
CALCulate[:SEARch|:TEST]:LIMit[:STATe]:MATCh:FILE: NAME:BASE . . . 341
CALCulate:ACPower:MARKer<x>:DELTa:X? (Query Only) . . . 342
CALCulate:ACPower:MARKer<x>:DELTa:Y? (Query Only) . . . 342
CALCulate:ACPower:MARKer<x>:MAXimum (No Query Form) . . . 343
CALCulate:ACPower:MARKer<x>:PEAK:LEFT (No Query Form) . . . 343
CALCulate:ACPower:MARKer<x>:PEAK:RIGHt (No Query Form) . . . 343
CALCulate:ACPower:MARKer<x>:X . . . 344
CALCulate:ACPower:MARKer<x>:Y? (Query Only) . . . 344
CALCulate:{AM|FM|PM}:MARKer<x>:DELTa:X? (Query Only) . . . 345
CALCulate:{AM|FM|PM}:MARKer<x>:DELTa:Y? (Query Only) . . . 345
CALCulate:{AM|FM|PM}:MARKer<x>:MAXimum (No Query Form) . . . 346
CALCulate:{AM|FM|PM}:MARKer<x>:PEAK:HIGHer (No Query Form) . . . 346
CALCulate:{AM|FM|PM}:MARKer<x>:PEAK:LEFT (No Query Form) . . . 347
CALCulate:{AM|FM|PM}:MARKer<x>:PEAK:LOWer (No Query Form) . . . 347
CALCulate:{AM|FM|PM}:MARKer<x>:PEAK:RIGHt (No Query Form) . . . 348
CALCulate:{AM|FM|PM}:MARKer<x>:X . . . 348
CALCulate:{AM|FM|PM}:MARKer<x>:Y? (Query Only) . . . 349
CALCulate:AUDio:HARMonic:HNUMber . . . 349
CALCulate:AUDio:HARMonic:NHNumber . . . 350
CALCulate:AUDio:NHARmonic:EXCursion . . . 350
CALCulate:AUDio:NHARmonic:IGNore . . . 350
CALCulate:AUDio:NHARmonic:THReshold . . . 351
CALCulate:AUDio:SPECtrum:MARKer<x>:DELTa:X? (Query Only) . . . 351
CALCulate:AUDio:SPECtrum:MARKer<x>:DELTa:Y? (Query Only) . . . 352
CALCulate:AUDio:SPECtrum:MARKer<x>:MAXimum (No Query Form) . . . 352
CALCulate:AUDio:SPECtrum:MARKer<x>:PEAK:HIGHer (No Query Form) . . . 353
CALCulate:AUDio:SPECtrum:MARKer<x>:PEAK:LEFT (No Query Form) . . . 353
CALCulate:AUDio:SPECtrum:MARKer<x>:PEAK:LOWer (No Query Form) . . . 354
CALCulate:AUDio:SPECtrum:MARKer<x>:PEAK:RIGHt (No Query Form) . . . 354
CALCulate:AUDio:SPECtrum:MARKer<x>:TRACe . . . 354
CALCulate:AUDio:SPECtrum:MARKer<x>:X . . . 355
CALCulate:AUDio:SPECtrum:MARKer<x>:Y? (Query Only) . . . 355
CALCulate:AVTime[:MEASview<y>]:MARKer<x>:DELTa:X? (Query Only) . . . 356
CALCulate:AVTime[:MEASview<y>]:MARKer<x>:DELTa:Y? (Query Only) . . . 357
CALCulate:AVTime[:MEASview<y>]:MARKer<x>:MAXimum (No Query Form) . . . 357

CALCulate:AVTime[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (No Query Form) 358
CALCulate:AVTime[:MEASview<y>]:MARKer<x>:PEAK:LEFT (No Query Form) 358
CALCulate:AVTime[:MEASview<y>]:MARKer<x>:PEAK:LOWer (No Query Form) 359
CALCulate:AVTime[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (No Query Form) 359
CALCulate:AVTime[:MEASview<y>]:MARKer<x>:TRACe 360
CALCulate:AVTime[:MEASview<y>]:MARKer<x>:X 360
CALCulate:AVTime[:MEASview<y>]:MARKer<x>:Y? (Query Only) 361
CALCulate:BLUEtooth:CONSte:MARKer<x>:DELTa:X[:TIME]? (Query only) 362
CALCulate:BLUEtooth:CONSte:MARKer<x>:FDEViation? (Query Only) 362
CALCulate:BLUETooth:CONSte:MARKer<x>:MAGNitude? (Query Only) 363
CALCulate:BLUEtooth:CONSte:MARKer<x>:MAXimum (No Query Form) 363
CALCulate:BLUEtooth:CONSte:MARKer<x>:PEAK:LEFT (No Query Form) 364
CALCulate:BLUEtooth:CONSte:MARKer<x>:PEAK:RIGHt (No Query Form) 364
CALCulate:BLUETooth:CONSte:MARKer<x>:PHASe? (Query Only) 365
CALCulate:BLUEtooth:CONSte:MARKer<x>:SYMBol? (Query Only) 365
CALCulate:BLUEtooth:CONSte:MARKer<x>:Trace 366
CALCulate:BLUETooth:CONSte:MARKer<x>:VALue? (Query Only) 366
CALCulate:BLUEtooth:CONSte:MARKer<x>:X 367
CALCulate:BLUEtooth:EDIagram:MARKer<x>:DELTa:X[:TIME]? (Query Only) 367
CALCulate:BLUEtooth:EDIagram:MARKer<x>:DELTa:Y? (Query Only) 368
CALCulate:BLUEtooth:EDIagram:MARKer<x>:MAXimum (No Query Form) 368
CALCulate:BLUEtooth:EDIagram:MARKer<x>:PEAK:HIGHer (No Query Form) 369
CALCulate:BLUEtooth:EDIagram:MARKer<x>:PEAK:LEFT (No Query Form) 369
CALCulate:BLUEtooth:EDIagram:MARKer<x>:PEAK:LOWer (No Query Form) 370
CALCulate:BLUEtooth:EDIagram:MARKer<x>:PEAK:RIGHt (No Query Form) 370
CALCulate:BLUEtooth:EDIagram:MARKer<x>:TRACe 371
CALCulate:BLUEtooth:EDIagram:MARKer<x>:X[:TIME] 371
CALCulate:BLUEtooth:EDIagram:MARKer<x>:Y? (Query Only) 372
CALCulate:BLUEtooth:FDVTIME:MARKer<x>:DELTa:X[:TIME]? (Query Only) 372
CALCulate:BLUEtooth:FDVTIME:MARKer<x>:DELTa:Y? (Query Only) 373
CALCulate:BLUEtooth:FDVTime:MARKer<x>:MAXimum (No Query Form) 374
CALCulate:BLUEtooth:FDVTime:MARKer<x>:PEAK:HIGHer (No Query Form) 374
CALCulate:BLUEtooth:FDVTime:MARKer<x>:PEAK:LEFT (No Query Form) 375
CALCulate:BLUEtooth:FDVTime:MARKer<x>:PEAK:LOWer (No Query Form) 375
CALCulate:BLUEtooth:FDVTime:MARKer<x>:PEAK:RIGHT (No Query Form) 376
CALCulate:BLUEtooth:FDVTIME:MARKer<x>:X[:TIME] 376
CALCulate:BLUEtooth:FDVTIME:MARKer<x>:Y 377
CALCulate:BOBW:MARKer<x>:DELTa:X? (Query Only) 377
CALCulate:BOBW:MARKer<x>:DELTa:Y? (Query Only) 378
CALCulate:BOBW:MARKer<x>:MAXimum (No Query Form) 378
CALCulate:BOBW:MARKer<x>:PEAK:HIGHer (No Query Form) 379
CALCulate:BOBW:MARKer<x>:PEAK:LEFT (No Query Form) 379
CALCulate:BOBW:MARKer<x>:PEAK:LOWer (No Query Form) 379
CALCulate:BOBW:MARKer<x>:PEAK:RIGHt (No Query Form) 380
CALCulate:BOBW:MARKer<x>[:SET]:CENTer (No Query Form) 380
CALCulate:BOBW:MARKer<x>:X 381

CALCulate:BOBW:MARKer<x>:Y? (Query Only) . . . 381
CALCulate:CONSte[:MEASview<y>]:MARKer<x>:DELTa:X[:TIME]? (Query only) . . . 382
CALCulate:CONSte[:MEASview<y>]:MARKer<x>:FDEViation? (Query only) . . . 382
CALCulate:CONSte[:MEASview<y>]:MARKer<x>:MAGNitude? (Query only) . . . 383
CALCulate:CONSte[:MEASview<y>]:MARKer<x>:MAXimum (No query form) . . . 384
CALCulate:CONSte[:MEASview<y>]:MARKer<x>:PEAK:LEFT (No query form) . . . 384
CALCulate:CONSte[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (No query form) . . . 385
CALCulate:CONSte[:MEASview<y>]:MARKer<x>:PHASe? (Query only) . . . 385
CALCulate:CONSte[:MEASview<y>]:MARKer<x>:SYMBol? (Query only) . . . 386
CALCulate:CONSte[:MEASview<y>]:MARKer<x>:VALue? (Query only) . . . 387
CALCulate:CONSte[:MEASview<y>]:MARKer<x>:X . . . 387
CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:DELTa:X[:TIME]? (Query only) . . . 388
CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:DELTa:Y? (Query only) . . . 389
CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:MAXimum (No query form) . . . 389
CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (No query form) . . . 390
CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:PEAK:LEFT (No query form) . . . 391
CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:PEAK:LOWer (No query form) . . . 391
CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (No query form) . . . 392
CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:TRACe . . . 392
CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:X[:TIME] . . . 393
CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:Y? (Query only) . . . 394
CALCulate:DPX:MARKer<x>:DELTa:X:AMPLitude? (Query Only) . . . 394
CALCulate:DPX:MARKer<x>:DELTa:X[:FREQuency]? (Query Only) . . . 395
CALCulate:DPX:MARKer<x>:DELTa:Y? (Query Only) . . . 395
CALCulate:DPX:MARKer<x>:MAXimum (No Query Form) . . . 396
CALCulate:DPX:MARKer<x>:PEAK:HIGHer (No Query Form) . . . 396
CALCulate:DPX:MARKer<x>:PEAK:LEFT (No Query Form) . . . 397
CALCulate:DPX:MARKer<x>:PEAK:LOWer (No Query Form) . . . 397
CALCulate:DPX:MARKer<x>:PEAK:RIGHt (No Query Form) . . . 398
CALCulate:DPX:MARKer<x>:POWer:DENSity? (Query Only) . . . 398
CALCulate:DPX:MARKer<x>:POWer:IDensity? (Query Only) . . . 399
CALCulate:DPX:MARKer<x>:POWer:INTEgrated? (Query Only) . . . 399
CALCulate:DPX:MARKer<x>[:SET]:CENTer (No Query Form) . . . 400
CALCulate:DPX:MARKer<x>:TRACe . . . 400
CALCulate:DPX:MARKer<x>:X:AMPLitude . . . 401
CALCulate:DPX:MARKer<x>:X[:FREQuency] . . . 401
CALCulate:DPX:MARKer<x>:X:PHASe . . . 402
CALCulate:DPX:MARKer<x>:X:TIME . . . 402
CALCulate:DPX:MARKer<x>:Y? (Query Only) . . . 403
CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:DELTa:X[:TIME]? (Query only) . . . 403
CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:DELTa:Y? (Query only) . . . 404
CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:MAXimum (No query form) . . . 405
CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (No query form) . . . 405
CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:PEAK:LEFT (No query form) . . . 406
CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:PEAK:LOWer (No query form) . . . 406
CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (No query form) . . . 407

CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:TRACe . . . 408
CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:X[:TIME] . . . 408
CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:Y? (Query only) . . . 409
CALCulate:EMC:MARKer<x>:DELTa:X? (Query Only) . . . 410
CALCulate:EMC:MARKer<x>:DELTa:Y? (Query Only) . . . 410
CALCulate:EMC:MARKer<x>:MAXimum (No Query Form) . . . 411
CALCulate:EMC:MARKer<x>:PEAK:HIGHer (No Query Form) . . . 411
CALCulate:EMC:MARKer<x>:PEAK:LEFT (No Query Form) . . . 411
CALCulate:EMC:MARKer<x>:PEAK:LOWer (No Query Form) . . . 412
CALCulate:EMC:MARKer<x>:PEAK:RIGHt (No Query Form) . . . 412
CALCulate:EMC:MARKer<x>[:SET]:CENTer (No Query Form) . . . 413
CALCulate:EMC:MARKer<x>:X . . . 413
CALCulate:EMC:MARKer<x>:Y . . . 414
CALCulate:EVM[:MEASview<y>]:MARKer<x>:DELTa:X? (Query only) . . . 414
CALCulate:EVM[:MEASview<y>]:MARKer<x>:DELTa:Y? (Query only) . . . 415
CALCulate:EVM[:MEASview<y>]:MARKer<x>:MAXimum (No query form) . . . 416
CALCulate:EVM[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (No query form) . . . 416
CALCulate:EVM[:MEASview<y>]:MARKer<x>:PEAK:LEFT (No query form) . . . 417
CALCulate:EVM[:MEASview<y>]:MARKer<x>:PEAK:LOWer (No query form) . . . 417
CALCulate:EVM[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (No query form) . . . 418
CALCulate:EVM[:MEASview<y>]:MARKer<x>:X . . . 419
CALCulate:EVM[:MEASview<y>]:MARKer<x>:Y? (Query only) . . . 419
CALCulate:FDVTime[:MEASview<y>]:MARKer<x>:DELTa:X[:TIME]? (Query only) . . . 420
CALCulate:FDVTime[:MEASview<y>]:MARKer<x>:DELTa:Y? (Query only) . . . 420
CALCulate:FDVTime[:MEASview<y>]:MARKer<x>:MAXimum (No query form) . . . 421
CALCulate:FDVTime[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (No query form) . . . 422
CALCulate:FDVTime[:MEASview<y>]:MARKer<x>:PEAK:LEFT (No query form) . . . 422
CALCulate:FDVTime[:MEASview<y>]:MARKer<x>:PEAK:LOWer (No query form) . . . 423
CALCulate:FDVTime[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (No query form) . . . 423
CALCulate:FDVTime[:MEASview<y>]:MARKer<x>:X[:TIME] . . . 424
CALCulate:FDVTime[:MEASview<y>]:MARKer<x>:Y? (Query only) . . . 425
CALCulate:{FSETtling|PSETtling}:MARKer<x>:DELTa:X? (Query only) . . . 425
CALCulate:{FSETtling|PSETtling}:MARKer<x>:DELTa:Y? (Query only) . . . 426
CALCulate:{FSETtling|PSETtling}:MARKer<x>:MAXimum (No Query Form) . . . 426
CALCulate:{FSETtling|PSETtling}:MARKer<x>:PEAK:HIGHer (No Query Form) . . . 427
CALCulate:{FSETtling|PSETtling}:MARKer<x>:PEAK:LEFT (No Query Form) . . . 427
CALCulate:{FSETtling|PSETtling}:MARKer<x>:PEAK:LOWer (No Query Form) . . . 428
CALCulate:{FSETtling|PSETtling}:MARKer<x>:PEAK:RIGHt (No Query Form) . . . 428
CALCulate:{FSETtling|PSETtling}:MARKer<x>:TRACe . . . 429
CALCulate:{FSETtling|PSETtling}:MARKer<x>:X . . . 429
CALCulate:{FSETtling|PSETtling}:MARKer<x>:Y? (Query Only) . . . 430
CALCulate:FVTime[:MEASview<y>]:MARKer<x>:DELTa:X? (Query Only) . . . 430
CALCulate:FVTime[:MEASview<y>]:MARKer<x>:DELTa:Y? (Query Only) . . . 431
CALCulate:FVTime[:MEASview<y>]:MARKer<x>:MAXimum (No Query Form) . . . 431
CALCulate:FVTime[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (No Query Form) . . . 432
CALCulate:FVTime[:MEASview<y>]:MARKer<x>:PEAK:LEFT (No Query Form) . . . 432

CALCulate:FVTime[:MEASview<y>]:MARKer<x>:PEAK:LOWer (No Query Form) 433
CALCulate:FVTime[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (No Query Form) 433
CALCulate:FVTime[:MEASview<y>]:MARKer<x>:TRACe 434
CALCulate:FVTime[:MEASview<y>]:MARKer<x>:X 434
CALCulate:FVTime[:MEASview<y>]:MARKer<x>:Y? (Query Only) 435
CALCulate:HARMonic:APPLy (No Query Form) 436
CALCulate:HARMonic:MARKer<x>:FACTor 436
CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:DELTa:X? (Query Only) 437
CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:DELTa:Y? (Query Only) 437
CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:MAXimum (No Query Form) 438
CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (No Query Form) 439
CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:PEAK:LEFT (No Query Form) 439
CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:PEAK:LOWer (No Query Form) 440
CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (No Query Form) 440
CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:TRACe 441
CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:X 441
CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:Y? (Query Only) 442
CALCulate:LTE:ACLR:MARKer:<x>PEAK:HIGHer (No Query Form) 442
CALCulate:LTE:ACLR:MARKer<x>:DELTa:X? (Query Only) 443
CALCulate:LTE:ACLR:MARKer<x>:DELTa:Y? (Query Only) 443
CALCulate:LTE:ACLR:MARKer<x>:MAXimum (No Query Form) 444
CALCulate:LTE:ACLR:MARKer<x>:PEAK:LEFT (No Query Form) 444
CALCulate:LTE:ACLR:MARKer<x>:PEAK:LOWer (No Query Form) 444
CALCulate:LTE:ACLR:MARKer<x>:PEAK:RIGHt (No Query Form) 445
CALCulate:LTE:ACLR:MARKer<x>:SET:CENTer (No Query Form) 445
CALCulate:LTE:ACLR:MARKer<x>:X 446
CALCulate:LTE:ACLR:MARKer<x>:Y? (Query Only) 446
CALCulate:LTE:CHSPectrum:MARKer<x>:DELTa:X? (Query Only) 447
CALCulate:LTE:CHSPectrum:MARKer<x>:DELTa:Y? (Query Only) 447
CALCulate:LTE:CHSPectrum:MARKer<x>:MAXimum (No Query Form) 447
CALCulate:LTE:CHSPectrum:MARKer<x>:PEAK:LEFT (No Query Form) 448
CALCulate:LTE:CHSPectrum:MARKer<x>:PEAK:RIGHt (No Query Form) 448
CALCulate:LTE:CHSPectrum:MARKer<x>:X 449
CALCulate:LTE:CHSPectrum:MARKer<x>:Y? (Query Only) 449
CALCulate:LTE:CONSte:MARKer<x>:FREQuency 450
CALCulate:LTE:CONSte:MARKer<x>:MAGNitude? (Query Only) 450
CALCulate:LTE:CONSte:MARKer<x>:PHASe? (Query Only) 451
CALCulate:LTE:CONSte:MARKer<x>:TIME 451
CALCulate:LTE:CONSte:MARKer<x>:TYPE? (Query Only) 452
CALCulate:LTE:PVTime:MARKer:<x>PEAK:HIGHer (No Query Form) 452
CALCulate:LTE:PVTime:MARKer<x>:DELTa:X? (Query Only) 452
CALCulate:LTE:PVTime:MARKer<x>:DELTa:Y? (Query Only) 453
CALCulate:LTE:PVTime:MARKer<x>:MAXimum (No Query Form) 453
CALCulate:LTE:PVTime:MARKer<x>:PEAK:LEFT (No Query Form) 454
CALCulate:LTE:PVTime:MARKer<x>:PEAK:LOWer (No Query Form) 454
CALCulate:LTE:PVTime:MARKer<x>:PEAK:RIGHt (No Query Form) 455

CALCulate:LTE:PVTime:MARKer<x>:X . . . 455
CALCulate:LTE:PVTime:MARKer<x>:Y? (Query Only) . . . 456
CALCulate:MARKer:ADD (No Query Form) . . . 456
CALCulate:MARKer:AOFF (No Query Form) . . . 457
CALCulate:MARKer:DELete (No Query Form) . . . 457
CALCulate:MARKer:DENSity:EXCursion . . . 458
CALCulate:MARKer:DENSity:SMOothing . . . 458
CALCulate:MARKer:DENSity:THReshold . . . 459
CALCulate:MARKer:DRAG:SEARch:STATe . . . 459
CALCulate:MARKer:MODE . . . 459
CALCulate:MARKer:MODE . . . 460
CALCulate:MARKer:NOISe:MODE . . . 460
CALCulate:MARKer:PEAK:EXCursion . . . 461
CALCulate:MARKer:PEAK:THReshold . . . 461
CALCulate:MCPower:MARKer<x>:DELTa:X? (Query Only) . . . 462
CALCulate:MCPower:MARKer<x>:DELTa:Y? (Query Only) . . . 462
CALCulate:MCPower:MARKer<x>:MAXimum (No Query Form) . . . 463
CALCulate:MCPower:MARKer<x>:PEAK:LEFT (No Query Form) . . . 463
CALCulate:MCPower:MARKer<x>:PEAK:RIGHt (No Query Form) . . . 464
CALCulate:MCPower:MARKer<x>:X . . . 464
CALCulate:MCPower:MARKer<x>:Y? (Query Only) . . . 464
CALCulate:MERRor[:MEASview<y>]:MARKer<x>:DELTa:X? (Query only) . . . 465
CALCulate:MERRor[:MEASview<y>]:MARKer<x>:DELTa:Y? (Query only) . . . 466
CALCulate:MERRor[:MEASview<y>]:MARKer<x>:MAXimum (No query form) . . . 466
CALCulate:MERRor[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (No query form) . . . 467
CALCulate:MERRor[:MEASview<y>]:MARKer<x>:PEAK:LEFT (No query form) . . . 467
CALCulate:MERRor[:MEASview<y>]:MARKer<x>:PEAK:LOWer (No query form) . . . 468
CALCulate:MERRor[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (No query form) . . . 469
CALCulate:MERRor[:MEASview<y>]:MARKer<x>:X . . . 469
CALCulate:MERRor[:MEASview<y>]:MARKer<x>:Y? (Query only) . . . 470
CALCulate:NOISe:FIGure:MARKer<x>:DELTa:X? (Query Only) . . . 470
CALCulate:NOISe:FIGure:MARKer<x>:DELTa:Y? (Query only) . . . 471
CALCulate:NOISe:FIGure:MARKer<x>:MAXimum (No Query Form) . . . 471
CALCulate:NOISe:FIGure:MARKer<x>:PEAK:HIGHer (No Query Form) . . . 472
CALCulate:NOISe:FIGure:MARKer<x>:PEAK:LEFT (No Query Form) . . . 472
CALCulate:NOISe:FIGure:MARKer<x>:PEAK:LOWer (No Query Form) . . . 473
CALCulate:NOISe:FIGure:MARKer<x>:PEAK:RIGHt (No Query Form) . . . 473
CALCulate:NOISe:FIGure:MARKer<x>:TRACe . . . 474
CALCulate:NOISe:FIGure:MARKer<x>:X? (Query Only) . . . 475
CALCulate:NOISe:FIGure:MARKer<x>:Y? (Query Only) . . . 475
CALCulate:NOISe:GAIN:MARKer<x>:DELTa:X? (Query Only) . . . 476
CALCulate:NOISe:GAIN:MARKer<x>:DELTa:Y? (Query Only) . . . 476
CALCulate:NOISe:GAIN:MARKer<x>:MAXimum? (Query Only) . . . 477
CALCulate:NOISe:GAIN:MARKer<x>:PEAK:HIGHer? (Query Only) . . . 477
CALCulate:NOISe:GAIN:MARKer<x>:PEAK:LEFT? (Query Only) . . . 478
CALCulate:NOISe:GAIN:MARKer<x>:PEAK:LOWer? (Query Only) . . . 478

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:RIGHt? (Query Only) 479
CALCulate:NOISe:GAIN:MARKer<x>:TRACe 479
CALCulate:NOISe:GAIN:MARKer<x>:X? (Query Only) 480
CALCulate:NOISe:GAIN:MARKer<x>:Y? (Query Only) 481
CALCulate:NOISe:TEMPerature:MARKer<x>:DELTa:X? (Query Only) 481
CALCulate:NOISe:TEMPerature:MARKer<x>:DELTa:Y? (Query Only) 482
CALCulate:NOISe:TEMPerature:MARKer<x>:MAXimum? (Query Only) 482
CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:HIGHer? (Query Only) 483
CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:LEFT? (Query Only) 483
CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:LOWer? (Query Only) 484
CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:RIGHt? (Query Only) 484
CALCulate:NOISe:TEMPerature:MARKer<x>:TRACe 485
CALCulate:NOISe:TEMPerature:MARKer<x>:X? (Query Only) 486
CALCulate:NOISe:TEMPerature:MARKer<x>:Y? (Query Only) 486
CALCulate:NOISe:YFACtor:MARKer<x>:DELTa:X? (Query Only) 487
CALCulate:NOISe:YFACtor:MARKer<x>:DELTa:Y? (Query Only) 487
CALCulate:NOISe:YFACtor:MARKer<x>:MAXimum (No Query Form) 488
CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:HIGHer? (Query Only) 488
CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:LEFT? (Query Only) 489
CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:LOWer? (Query Only) 489
CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:RIGHt? (Query Only) 490
CALCulate:NOISe:YFACtor:MARKer<x>:TRACe 490
CALCulate:NOISe:YFACtor:MARKer<x>:X? (Query Only) 491
CALCulate:NOISe:YFACtor:MARKer<x>:Y? (Query Only) 491
CALCulate:OBWidth:MARKer<x>:DELTa:X? (Query Only) 492
CALCulate:OBWidth:MARKer<x>:DELTa:Y? (Query Only) 492
CALCulate:OBWidth:MARKer<x>:MAXimum (No Query Form) 493
CALCulate:OBWidth:MARKer<x>:PEAK:HIGHer (No Query Form) 493
CALCulate:OBWidth:MARKer<x>:PEAK:LEFT (No Query Form) 494
CALCulate:OBWidth:MARKer<x>:PEAK:LOWer (No Query Form) 494
CALCulate:OBWidth:MARKer<x>:PEAK:RIGHt (No Query Form) 495
CALCulate:OBWidth:MARKer<x>[:SET]:CENTer (No Query Form) 495
CALCulate:OBWidth:MARKer<x>:X 495
CALCulate:OBWidth:MARKer<x>:Y? (Query Only) 496
CALCulate:OFDM:CONSte:MARKer<x>:FREQuency 496
CALCulate:OFDM:CONSte:MARKer<x>:MAGNitude? (Query Only) 497
CALCulate:OFDM:CONSte:MARKer<x>:PHASe? (Query Only) 497
CALCulate:OFDM:CONSte:MARKer<x>:TIME 498
CALCulate:OFDM:CONSte:MARKer<x>:TYPE? (Query Only) 498
CALCulate:OFDM:CONSte:MARKer<x>:VALue? (Query Only) 499
CALCulate:OFDM:CRESponse:MARKer<x>:DELTa:X? (Query Only) 499
CALCulate:OFDM:CRESponse:MARKer<x>:DELTa:Y? (Query Only) 500
CALCulate:OFDM:CRESponse:MARKer<x>:MAXimum (No Query Form) 500
CALCulate:OFDM:CRESponse:MARKer<x>:PEAK:HIGHer (No Query Form) 500
CALCulate:OFDM:CRESponse:MARKer<x>:PEAK:LEFT (No Query Form) 501
CALCulate:OFDM:CRESponse:MARKer<x>:PEAK:LOWer (No Query Form) 501

CALCulate:OFDM:CRESponse:MARKer<x>:PEAK:RIGHt (No Query Form) 502
CALCulate:OFDM:CRESponse:MARKer<x>:X 502
CALCulate:OFDM:CRESponse:MARKer<x>:Y? (Query Only) 503
CALCulate:OFDM:FLATness:MARKer<x>:MAXimum (No Query Form) 503
CALCulate:OFDM:FLATness:MARKer<x>:PEAK:HIGHer (No Query Form) 503
CALCulate:OFDM:FLATness:MARKer<x>:PEAK:LEFT (No Query Form) 504
CALCulate:OFDM:FLATness:MARKer<x>:PEAK:LOWer (No Query Form) 504
CALCulate:OFDM:FLATness:MARKer<x>:PEAK:RIGHt (No Query Form) 505
CALCulate:OFDM:FLATness:MARKer<x>:X 505
CALCulate:OFDM:FLATness:MARKer<x>:Y? (Query Only) 506
CALCulate:OFDM:TABLe:MARKer<x>:FREQuency 506
CALCulate:OFDM:TABLe:MARKer<x>:TIMe 506
CALCulate:OFDM:TABLe:MARKer<x>:VALue? (Query Only) 507
CALCulate:P25:CONSte:MARKer<x>:DELTa:X[:TIME]? (Query Only) 507
CALCulate:P25:CONSte:MARKer<x>:FDEViation? (Query Only) 508
CALCulate:P25:CONSte:MARKer<x>:MAGNitude? (Query Only) 508
CALCulate:P25:CONSte:MARKer<x>:MAXimum (No Query Form) 509
CALCulate:P25:CONSte:MARKer<x>:PEAK:LEFT (No Query Form) 509
CALCulate:P25:CONSte:MARKer<x>:PEAK:RIGHt (No Query Form) 509
CALCulate:P25:CONSte:MARKer<x>:PHASe? (Query Only) 510
CALCulate:P25:CONSte:MARKer<x>:SYMBol? (Query Only) 510
CALCulate:P25:CONSte:MARKer<x>:Trace 511
CALCulate:P25:CONSte:MARKer<x>:VALue? (Query Only) 511
CALCulate:P25:CONSte:MARKer<x>:X 512
CALCulate:P25:EDIagram:MARKer<x>:DELTa:X[:TIME]? (Query Only) 512
CALCulate:P25:EDIagram:MARKer<x>:DELTa:Y? (Query Only) 513
CALCulate:P25:EDIagram:MARKer<x>:MAXimum (No Query Form) 513
CALCulate:P25:EDIagram:MARKer<x>:PEAK:HIGHer (No Query Form) 514
CALCulate:P25:EDIagram:MARKer<x>:PEAK:LEFT (No Query Form) 515
CALCulate:P25:EDIagram:MARKer<x>:PEAK:LOWer (No Query Form) 515
CALCulate:P25:EDIagram:MARKer<x>:PEAK:RIGHt (No Query Form) 516
CALCulate:P25:EDIagram:MARKer<x>:TRACe 517
CALCulate:P25:EDIagram:MARKer<x>:X 517
CALCulate:P25:EDIagram:MARKer<x>:Y 518
CALCulate:P25:PVTime:MARKer<x>:DELTa:X[:TIME]? (Query Only) 518
CALCulate:P25:PVTime:MARKer<x>:DELTa:Y[:TIME]? (Query Only) 519
CALCulate:P25:PVTime:MARKer<x>:Maximum (No Query Form) 519
CALCulate:P25:PVTime:MARKer<x>:PEAK:HIGHer (No Query Form) 520
CALCulate:P25:PVTime:MARKer<x>:PEAK:LEFT (No Query Form) 521
CALCulate:P25:PVTime:MARKer<x>:PEAK:LOWer (No Query Form) 521
CALCulate:P25:PVTime:MARKer<x>:PEAK:RIGHt (No Query Form) 522
CALCulate:P25:PVTime:MARKer<x>:X 522
CALCulate:P25:PVTime:MARKer<x>:Y 523
CALCulate:PERRor[:MEASview<y>]:MARKer<x>:DELTa:X? (Query only) 523
CALCulate:PERRor[:MEASview<y>]:MARKer<x>:DELTa:Y? (Query only) 524
CALCulate:PERRor[:MEASview<y>]:MARKer<x>:MAXimum (No query form) 525

CALCulate:PERRor[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (No query form) 525
CALCulate:PERRor[:MEASview<y>]:MARKer<x>:PEAK:LEFT (No query form) 526
CALCulate:PERRor[:MEASview<y>]:MARKer<x>:PEAK:LOWer (No query form) 526
CALCulate:PERRor[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (No query form) 527
CALCulate:PERRor[:MEASview<y>]:MARKer<x>:X 528
CALCulate:PERRor[:MEASview<y>] :MARKer<x>:Y? (Query only) 528
CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:DELTa:X? (Query Only) 529
CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:DELTa:Y? (Query Only) 529
CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:MAXimum (No Query Form) 530
CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (No Query Form) 530
CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:PEAK:LEFT (No Query Form) 531
CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:PEAK:LOWer (No Query Form) 531
CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (No Query Form) 532
CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:TRACe 533
CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:X 533
CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:Y? (Query Only) 534
CALCulate:PULSe:CUMulative:HISTogram[:MEASview<y>]:INDicator:X 534
CALCulate:PULSe:CUMulative:HISTogram[:MEASview<y>]:INDicator:Y (Query Only) 535
CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:FVTime: DELTa:X (Query Only) 535
CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:FVTime: DELTa:Y (Query Only) 536
CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:FVTime: MAXimum (No Query Form) 536
CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:FVTime:X 537
CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:FVTime:Y (Query Only) 537
CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:TVTime: DELTa:X (Query Only) 538
CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:TVTime: DELTa:Y (Query Only) 538
CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>: TVTime:MAXimum (No Query Form) 539
CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:TVTime:X 539
CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:TVTime:Y (Query Only) 540
CALCulate:PULSe:STATistics[:MEASview<y>]:FFT:INDicator:X 541
CALCulate:PULSe:STATistics[:MEASview<y>]:FFT:INDicator:Y? (Query)Only 541
CALCulate:PULSe:STATistics[:MEASview<y>]:HISTogram:INDicator:X 542
CALCulate:PULSe:STATistics[:MEASview<y>]:HISTogram:INDicator:Y? (Query Only) 542
CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:DELTa:X? 543
CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:DELTa:Y? 543
CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:MAXimum 544
CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>: PEAK:HIGHer 544
CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:PEAK:LEFT 545
CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>: PEAK:LOWer 546
CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>: PEAK:RIGHt 546
CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:X 547
CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:Y? 547
CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:DELTa:X 548
CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:DELTa:Y 548
CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:MAXimum 549
CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:PEAK:HIGHer 549
CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:PEAK:LEFT 550

CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:PEAK:LOWer 550
CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:PEAK:RIGHt 551
CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:X 551
CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:Y 552
CALCulate:SEARch:LIMit:FAIL? (Query Only) 553
CALCulate:SEARch:LIMit:MATCh:BEEP[:STATe] 553
CALCulate:SEARch:LIMit:MATCh:SACQuire[:STATe] 553
CALCulate:SEARch:LIMit:MATCh:SDATa[:STATe] 554
CALCulate:SEARch:LIMit:MATCh:SPICture[:STATe] 554
CALCulate:SEARch:LIMit:MATCh:STRace[:STATe] 555
CALCulate:SEARch:LIMit:OPERation 556
CALCulate:SEARch:LIMit:OPERation:FEED 557
CALCulate:SEARch:LIMit:OPERation:MASK:LOAD (No Query Form) 557
CALCulate:SEARch:LIMit:OPERation:MASK:STORE (No Query Form) 558
CALCulate:SEARch:LIMit:OPERation:SLIMit 558
CALCulate:SEARch:LIMit:REPort:DATA? (Query Only) 559
CALCulate:SEARch:LIMit:REPort:POINts? (Query Only) 559
CALCulate:SEARch:LIMit:STATe 560
CALCulate:SEARch:LIMit:STATe:MATCh:ACQuisition:FORMat 560
CALCulate:SEARch:LIMit:STATe:MATCh:BEEP:STATe 560
CALCulate:SEARch:LIMit:STATe:MATCh:COUNt:STATe 561
CALCulate:SEARch:LIMit:STATe:MATCh:COUNt:STATe:MAXimum 561
CALCulate:SEARch:LIMit:STATe:MATCh:FILE:LOCation 562
CALCulate:SEARch:LIMit:STATe:MATCh:FILE:NAME:BASE 562
CALCulate:SEARch:LIMit:STATe:MATCh:PICTure:FORMat 563
CALCulate:SEARch:LIMit:STATe:MATCh:SACQuire:STATe 563
CALCulate:SEARch:LIMit:STATe:MATCh:SDATa:STATe 564
CALCulate:SEARch:LIMit:STATe:MATCh:SPICture:STATe 564
CALCulate:SEARch:LIMit:STATe:MATCh:STRace:STATe 565
CALCulate:SEARch:LIMit:STATe:OPERation 565
CALCulate:SEARch:LIMit:STATe:OPERation:FEED 566
CALCulate:SEARch:LIMit:STATe:OPERation:MASK:LOAD (No Query Form) 567
CALCulate:SEARch:LIMit:STATe:OPERation:MASK:LOWer:STATe 567
CALCulate:SEARch:LIMit:STATe:OPERation:MASK:NEW:AUTO (No Query Form) 568
CALCulate:SEARch:LIMit:STATe:OPERation:MASK:STORE (No Query Form) 569
CALCulate:SEARch:LIMit:STATe:OPERation:MASK:UPPer:STATe 569
CALCulate:SEARch:LIMit:STATe 570
CALCulate:SEM:MARKer<x>:DELTa:X? (Query Only) 570
CALCulate:SEM:MARKer<x>:DELTa:Y? (Query Only) 571
CALCulate:SEM:MARKer<x>:MAXimum (No Query Form) 571
CALCulate:SEM:MARKer<x>:PEAK:HIGHer (No Query Form) 571
CALCulate:SEM:MARKer<x>:PEAK:LEFT (No Query Form) 572
CALCulate:SEM:MARKer<x>:PEAK:LOWer (No Query Form) 572
CALCulate:SEM:MARKer<x>:PEAK:RIGHt (No Query Form) 573
CALCulate:SEM:MARKer<x>[:SET]:CENTer (No Query Form) 573
CALCulate:SEM:MARKer<x>:X 573

CALCulate:SEM:MARKer<x>:Y? (Query Only) . . . 574
CALCulate:SGRam[:MEASview<y>]:MARKer<x>:DELTa:X: FREQuency . . . 574
CALCulate:SGRam[:MEASview<y>]:MARKer<x>:DELTa:X[:TIME] (Query Only) . . . 575
CALCulate:SGRam[:MEASview<y>]:MARKer<x>:DELTa:Y (Query Only) . . . 575
CALCulate:SGRam[:MEASview<y>]:MARKer<x>:MAXimum (No Query Form) . . . 576
CALCulate:SGRam[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (No Query Form) . . . 576
CALCulate:SGRam[:MEASview<y>]:MARKer<x>:PEAK:LEFT (No Query Form) . . . 577
CALCulate:SGRam[:MEASview<y>]:MARKer<x>:PEAK:LOWer (No Query Form) . . . 577
CALCulate:SGRam[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (No Query Form) . . . 578
CALCulate:SGRam[:MEASview<y>]:MARKer<x>[:SET]:CENTer (No Query Form) . . . 579
CALCulate:SGRam[:MEASview<y>]:MARKer<x>:X:FREQuency . . . 579
CALCulate:SGRam[:MEASview<y>]:MARKer<x>:X[:TIME] . . . 580
CALCulate:SGRam[:MEASview<y>]:MARKer<x>:Y (Query Only) . . . 580
CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:DELTa:X? (Query Only) . . . 581
CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:DELTa:Y? (Query Only) . . . 581
CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:MAXimum . . . 582
CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (No Query Form) . . . 582
CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:PEAK:LEFT (No Query Form) . . . 583
CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:PEAK:LOWer (No Query Form) . . . 583
CALCulate:SPECtrum[:MEASview<y>]:MARKer<x:PEAK:RIGHt (No Query Form) . . . 584
CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:POWer:DENSity? (Query Only) . . . 584
CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:POWer:IDENSity? . . . 585
CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:POWer:INTEgrated? (Query Only) . . . 586
CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>[:SET]:CENTer (No Query Form) . . . 586
CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:TRACe . . . 587
CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:X . . . 587
CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:Y? (Query Only) . . . 588
CALCulate:SPURious:MARKer<x>:DELTa:X? (Query Only) . . . 589
CALCulate:SPURious:MARKer<x>:DELTa:Y? (Query Only) . . . 589
CALCulate:SPURious:MARKer<x>:MAXimum (No Query Form) . . . 590
CALCulate:SPURious:MARKer<x>:PEAK:HIGHer (No Query Form) . . . 590
CALCulate:SPURious:MARKer<x>:PEAK:LEFT (No Query Form) . . . 590
CALCulate:SPURious:MARKer<x>:PEAK:LOWer (No Query Form) . . . 591
CALCulate:SPURious:MARKer<x>:PEAK:RIGHt (No Query Form) . . . 591
CALCulate:SPURious:MARKer<x>[:SET]:CENTer (No Query Form) . . . 591
CALCulate:SPURious:MARKer<x>:X . . . 592
CALCulate:SPURious:MARKer<x>:Y? (Query Only) . . . 592
CALCulate:SYNC:MARKer<x>:MEASurement . . . 593
CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:DELTa:X[:TIME]? (Query only) . . . 593
CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:DELTa:Y? (Query only) . . . 594
CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:MAXimum (No query form) . . . 595
CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (No query form) . . . 595
CALCulate:TDIagram[:MEASview<y>] :MARKer<x>:PEAK:LEFT (No query form) . . . 596
CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:PEAK:LOWer (No query form) . . . 596
CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (No query form) . . . 597
CALCulate:TDIagram [:MEASview<y>]:MARKer<x>:X[:TIME] . . . 597

CALCulate:TDIagram [:MEASview<y>]:MARKer<x>:Y? (Query only) 598
CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:DELTa:X? (Query Only) 599
CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:DELTa:Y? (Query Only) 599
CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:MAXimum (No Query Form) 600
CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (No Query Form) 600
CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:PEAK:LEFT (No Query Form) 601
CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:PEAK:LOWer (No Query Form) 601
CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (No Query Form) 602
CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:X 602
CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:Y? (Query Only) 603
CALculate:TXGain:MARKer<x>:DELTa:X? (Query Only) 604
CALculate:TXGain:MARKer<x>:DELTa:Y? (Query Only) 604
CALculate:TXGain:MARKer<x>:MAXimum (No Query Form) 604
CALculate:TXGain:MARKer<x>:PEAK:HIGHer (No Query Form) 605
CALculate:TXGain:MARKer<x>:PEAK:LEFT (No Query Form) 605
CALculate:TXGain:MARKer<x>:PEAK:LOWer (No Query Form) 606
CALculate:TXGain:MARKer<x>:PEAK:RIGHt (No Query Form) 606
CALculate:TXGain:MARKer<x>:TRACe 607
CALculate:TXGain:MARKer<x>:X 607
CALculate:TXGain:MARKer<x>:Y? (Query Only) 608
CALCulate:WLAN:AD:CONSte:MARKer<x>:DELTa:X[:TIME]? (Query Only) 608
CALCulate:WLAN:AD:CONSte:MARKer<x>:MAGNitude? (Query Only) 609
CALCulate:WLAN:AD:CONSte:MARKer<x>:PHASe? (Query Only) 609
CALCulate:WLAN:AD:CONSte:MARKer<x>:SYMBol? (Query Only) 609
CALCulate:WLAN:AD:CONSte:MARKer<x>:TYPE? (Query Only) 610
CALCulate:WLAN:AD:CONSte:MARKer<x>:VALue? (Query Only) 610
CALCulate:WLAN:AD:CONSte:MARKer<x>:X 611
CALCulate:WLAN:AD:EVM:MARKer<x>:DELTa:X? (Query Only) 611
CALCulate:WLAN:AD:EVM:MARKer<x>:DELTa:Y? (Query Only) 612
CALCulate:WLAN:AD:EVM:MARKer<x>:MAXimum (No Query Form) 612
CALCulate:WLAN:AD:EVM:MARKer<x>:PEAK:HIGHer (No Query Form) 613
CALCulate:WLAN:AD:EVM:MARKer<x>:PEAK:LEFT (No Query Form) 613
CALCulate:WLAN:AD:EVM:MARKer<x>:PEAK:LOWer (No Query Form) 614
CALCulate:WLAN:AD:EVM:MARKer<x>:PEAK:RIGHt (No Query Form) 614
CALCulate:WLAN:AD:EVM:MARKer<x>:X 615
CALCulate:WLAN:AD:EVM:MARKer<x>:Y? (Query Only) 615
CALCulate:WLAN:AY:CONSte:MARKer<x>:DELTa:X:[:TIME]? (Query Only) 616
CALCulate:WLAN:AY:CONSte:MARKer<x>:MAGNitude? (Query Only) 616
CALCulate:WLAN:AY:CONSte:MARKer<x>:PHASe? (Query Only) 617
CALCulate:WLAN:AY:CONSte:MARKer<x>:SYMBol? (Query Only) 617
CALCulate:WLAN:AY:CONSte:MARKer<x>:TYPE? (Query Only) 617
CALCulate:WLAN:AY:CONSte:MARKer<x>:VALue? (Query Only) 618
CALCulate:WLAN:AY:CONSte:MARKer<x>:X 618
CALCulate:WLAN:AY:EVM:MARKer<x>:DELTa:X 619
CALCulate:WLAN:AY:EVM:MARKer<x>:DELTa:Y 619
CALCulate:WLAN:AY:EVM:MARKer<x>:MAXimum (No Query Form) 620

CALCulate:WLAN:AY:EVM:MARKer<x>:PEAK:HIGHer (No Query Form) 621
CALCulate:WLAN:AY:EVM:MARKer<x>:PEAK:LEFT (No Query Form) 621
CALCulate:WLAN:AY:EVM:MARKer<x>:PEAK:LOWer (No Query Form) 622
CALCulate:WLAN:AY:EVM:MARKer<x>:PEAK:RIGHt (No Query Form) 622
CALCulate:WLAN:AY:EVM:MARKer<x>:X 623
CALCulate:WLAN:AY:EVM:MARKer<x>:Y 623
CALCulate:WLAN:CONSte:MARKer<x>:FREQuency 624
CALCulate:WLAN:CONSte:MARKer<x>:MAGNitude? (Query Only) 625
CALCulate:WLAN:CONSte:MARKer<x>:PHASe? (Query Only) 625
CALCulate:WLAN:CONSte:MARKer<x>:TIME 626
CALCulate:WLAN:CONSte:MARKer<x>:TYPE? (Query Only) 626
CALCulate:WLAN:CONSte:MARKer<x>:VALue? (Query Only) 627
CALCulate:WLAN:CRESponse:MARKer<x>:DELTa:X? (Query Only) 627
CALCulate:WLAN:CRESponse:MARKer<x>:DELTa:Y? (Query Only) 628
CALCulate:WLAN:CRESponse:MARKer<x>:MAXimum (No Query Form) 628
CALCulate:WLAN:CRESponse:MARKer<x>:PEAK:HIGHer (No Query Form) 629
CALCulate:WLAN:CRESponse:MARKer<x>:PEAK:LEFT (No Query Form) 629
CALCulate:WLAN:CRESponse:MARKer<x>:PEAK:LOWer (No Query Form) 630
CALCulate:WLAN:CRESponse:MARKer<x>:PEAK:RIGHt (No Query Form) 630
CALCulate:WLAN:CRESponse:MARKer<x>:X 631
CALCulate:WLAN:CRESponse:MARKer<x>:Y? (Query Only) 631
CALCulate:WLAN:EVM:MARKer<x>:AVERage:SUBCarrier? (Query Only) 632
CALCulate:WLAN:EVM:MARKer<x>:AVERage:SYMBol? (Query Only) 632
CALCulate:WLAN:EVM:MARKer<x>:DELTa:SUBCarrier? (Query Only) 633
CALCulate:WLAN:EVM:MARKer<x>:DELTa:SYMBol? (Query Only) 633
CALCulate:WLAN:EVM:MARKer<x>:DELTa:Y? (Query Only) 634
CALCulate:WLAN:EVM:MARKer<x>:FREQuency 634
CALCulate:WLAN:EVM:MARKer<x>:TIME 635
CALCulate:WLAN:EVM:MARKer<x>:VALue? (Query Only) 636
CALCulate:WLAN:FLATness:MARKer<x>:MAXimum (No Query Form) 636
CALCulate:WLAN:FLATness:MARKer<x>:PEAK:HIGHer (No Query Form) 637
CALCulate:WLAN:FLATness:MARKer<x>:PEAK:LEFT (No Query Form) 637
CALCulate:WLAN:FLATness:MARKer<x>:PEAK:LOWer (No Query Form) 638
CALCulate:WLAN:FLATness:MARKer<x>:PEAK:RIGHt (No Query Form) 638
CALCulate:WLAN:FLATness:MARKer<x>:X 639
CALCulate:WLAN:FLATness:MARKer<x>:Y? (Query Only) 639
CALCulate:WLAN:MERRor:MARKer<x>:AVERage:SUBCarrier? (Query Only) 640
CALCulate:WLAN:MERRor:MARKer<x>:AVERage:SYMBol? (Query Only) 640
CALCulate:WLAN:MERRor:MARKer<x>:DELTa:SUBCarrier? (Query Only) 641
CALCulate:WLAN:MERRor:MARKer<x>:DELTa:SYMBol? (Query Only) 641
CALCulate:WLAN:MERRor:MARKer<x>:DELTa:Y? (Query Only) 642
CALCulate:WLAN:MERRor:MARKer<x>:FREQuency 643
CALCulate:WLAN:MERRor:MARKer<x>:TIME 643
CALCulate:WLAN:MERRor:MARKer<x>:VALue? (Query Only) 644
CALCulate:WLAN:PERRor:MARKer<x>:AVERage:SUBCarrier? (Query Only) 645
CALCulate:WLAN:PERRor:MARKer<x>:AVERage:SYMBol? (Query Only) 645

CALCulate:WLAN:PERRor:MARKer<x>:DELTa:SUBCarrier? (Query Only) 646
CALCulate:WLAN:PERRor:MARKer<x>:DELTa:SYMBol? (Query Only) 646
CALCulate:WLAN:PERRor:MARKer<x>:DELTa:Y? (Query Only) 647
CALCulate:WLAN:PERRor:MARKer<x>:FREQuency 647
CALCulate:WLAN:PERRor:MARKer<x>:TIME 648
CALCulate:WLAN:PERRor:MARKer<x>:VALue? (Query Only) 649
CALCulate:WLAN:PVTime:MARKer<x>:DELTa:X? (Query Only) 649
CALCulate:WLAN:PVTime:MARKer<x>:DELTa:Y? (Query Only) 650
CALCulate:WLAN:PVTime:MARKer<x>:MAXimum (No Query Form) 650
CALCulate:WLAN:PVTime:MARKer<x>:PEAK:HIGHer (No Query Form) 651
CALCulate:WLAN:PVTime:MARKer<x>:PEAK:LEFT (No Query Form) 651
CALCulate:WLAN:PVTime:MARKer<x>:PEAK:LOWer (No Query Form) 652
CALCulate:WLAN:PVTime:MARKer<x>:PEAK:RIGHt (No Query Form) 652
CALCulate:WLAN:PVTime:MARKer<x>:X 652
CALCulate:WLAN:PVTime:MARKer<x>:Y? (Query Only) 653
CALCulate:WLAN:TABLe:MARKer<x>:FREQuency 653
CALCulate:WLAN:TABLe:MARKer<x>:TIME 654
CALCulate:WLAN:TABLe:MARKer<x>:VALue? (Query Only) 655
Calibration commands 655
CALibration:ABORt (No query form) 655
CALibration:AUTO 656
CALibration:CORRection:EXTernal:EDIT<x>:LABel 656
CALibration:CORRection:EXTernal:EDIT<x>:NEW (No query form) 656
CALibration:CORRection:EXTernal:EDIT<x>:STATe 657
CALibration:CORRection:EXTernal:GAIN[:MAGNitude] 658
CALibration:CORRection:EXTernal:GAIN:STATe 658
CALibration:CORRection:EXTernal:PROBe:CONNect? (Query only) 659
CALibration:CORRection:EXTernal:PROBe[:MAGNitude]? (Query only) 659
CALibration:CORRection:EXTernal:PROBe:STATe 660
CALibration:CORRection:EXTernal:TYPE 660
CALibration:TXGain:ABORt (No query form) 661
CALibration:TXGain[:ALL] (No query form) 661
CALibration:TXGain:FINish? (Query only) 661
CALibration:TXGain:STATe? (Query only) 662
*CLS (No Query Form) 662
Display commands 663
DISPlay:ACPower:MARKer:SHOW:STATe 663
DISPlay:ACPower:PLEVel:SHOW:STATe 664
DISPlay:ACPower:RESet:SCALe (No Query Form) 664
DISPlay:ACPower:WINDow:TRACe:GRATicule:GRID:STATe 664
DISPlay:ACPower:X[:SCALe] 665
DISPlay:ACPower:X[:SCALe]:AUTO (No Query Form) 665
DISPlay:ACPower:X[:SCALe]:OFFSet 666
DISPlay:ACPower:Y[:SCALe] 666
DISPlay:ACPower:Y[:SCALe]:AUTO (No Query Form) 667
DISPlay:ACPower:Y[:SCALe]:OFFSet 667

DISPlay:ADEMod:MEASview:DELete (No Query Form) 668
DISPlay:ADEMod:MEASview:NEW (No Query Form) 668
DISPlay:ADEMod:MEASview:SELect 669
DISPlay:{AM|FM|PM}:MARKer:SHOW:STATe 669
DISPlay:{AM|FM|PM}:WINDow:TRACe:GRATicule:GRID:STATe 670
DISPlay:{AM|FM|PM}:X:RSCale (No Query Form) 670
DISPlay:{AM|FM|PM}:X[:SCALe]:AUTO (No Query Form) 671
DISPlay:{AM|FM|PM}:X[:SCALe]:FULL 671
DISPlay:{AM|FM|PM}:X[:SCALe]:OFFSet 671
DISPlay:{AM|FM|PM}:Y:RSCale (No Query Form) 672
DISPlay:{AM|FM|PM}:Y[:SCALe] 672
DISPlay:{AM|FM|PM}:Y[:SCALe]:OFFSet 673
DISPlay:ANTenna:AZIMuth:SHOW:STATe 674
DISPlay:AUDio:MEASview:DELete (No Query Form) 674
DISPlay:AUDio:MEASview:NEW (No Query Form) 674
DISPlay:AUDio:MEASview:SELect 675
DISPlay:AUDio:SPECtrum:FREQuency:AUTO (No Query Form) 675
DISPlay:AUDio:SPECtrum:FREQuency:[SCALe]:STARt 676
DISPlay:AUDio:SPECtrum:FREQuency:[SCALe]:STOP 676
DISPlay:AUDio:SPECtrum:MARKer:SHOW:STATe 677
DISPlay:AUDio:SPECtrum:RESet:SCALe (No Query Form) 677
DISPlay:AUDio:SPECtrum:SCALe:LOG:STATe 677
DISPlay:AUDio:SPECtrum:SHOW:NHARmonic:THReshold 678
DISPlay:AUDio:SPECtrum:TABLe:SHOW:STATe 678
DISPlay:AUDio:SPECtrum:WINDow:TRACe:GRATicule:GRID:STATe 679
DISPlay:AUDio:SPECtrum:Y:[SCALe] 679
DISPlay:AUDio:SPECtrum:Y:[SCALe]:AUTO (No Query Form) 680
DISPlay:AUDio:SPECtrum:Y:[SCALe]:OFFSet 680
DISPlay:AVTime[:MEASview<y>]:LEGend:STATe 680
DISPlay:AVTime[:MEASview<y>]:MARKer:SHOW:STATe 681
DISPlay:AVTime[:MEASview<y>]:SELect:FIRSt 681
DISPlay:AVTime[:MEASview<y>]:SELect:SECond 682
DISPlay:AVTime[:MEASview<y>]:SELect:THIRd 682
DISPlay:AVTime[:MEASview<y>]:SELect:FOURth 683
DISPlay:AVTime[:MEASview<y>]:SELect:TIMestamp 684
DISPlay:AVTime[:MEASview<y>]:READout[:STATe] 684
DISPlay:AVTime[:MEASview<y>]:RESet (No Query Form) 685
DISPlay:AVTime[:MEASview<y>]:TRIGger:LEVel:STATe 685
DISPlay:AVTime[:MEASview<y>]:WINDow:TRACe:GRATicule: GRID:STATe 686
DISPlay:AVTime[:MEASview<y>]:X:RSCale (No Query Form) 686
DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:AUTO (No Query Form) 687
DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:AUTO:STATe 687
DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:FULL 688
DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:MAXimum? (Query Only) 688
DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:MINimum? (Query Only) 689
DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:OFFSet 689

DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MAXimum? (Query Only) 690
DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MINimum? (Query Only) 691
DISPlay:AVTime[:MEASview<y>]:Y:RSCale (No Query Form) 691
DISPlay:AVTime[:MEASview<y>]:Y[:SCALe]:FULL 692
DISPlay:AVTime[:MEASview<y>]:Y[:SCALe]:OFFSet 692
DISPlay:BIBEmissions:MARKer<x>:SHOW:STATe 693
DISPlay:BIBEmissions:PLEVel:SHOW:STATe 693
DISPlay:BIBEmissions:RESet:SCALe (No Query Form) 694
DISPlay:BIBEmissions:WINDow:TRACe:GRATicule:GRID:STATe 694
DISPlay:BIBEmissions:X[:SCALe] 695
DISPlay:BIBEmissions:X[:SCALe]:AUTO (No Query Form) 695
DISPlay:BIBEmissions:X[:SCALe]:OFFSet 696
DISPlay:BIBEmissions:Y[:SCALe] 696
DISPlay:BIBEmissions:Y[:SCALe]:AUTO (No Query Form) 697
DISPlay:BIBEmissions:Y[:SCALe]:OFFSet 697
DISPlay:BLUEtooth:CONSte:TRACe:GRATicule:GRID:STATe 698
DISPlay:BLUEtooth:CONSte:WINDow:TRACe:GRATicule:GRID:STATe 698
DISPlay:BLUEtooth:EDIagram:WINDow:TRACe:GRATicule:GRID:STATe 699
DISPlay:BLUEtooth:EDIagram:Y[:SCALe] 699
DISPlay:BLUEtooth:EDIagram:Y[:SCALe]:AUTO (No Query Form) 700
DISPlay:BLUEtooth:EDIagram:Y[:SCALe]:OFFSet 700
DISPlay:BLUEtooth:FDVTime:WINDow:TRACe:GRATicule:GRID:STATe 700
DISPlay:BLUEtooth:FDVTime:Y[:SCALe]:AUTO (No Query Form) 701
DISPlay:BLUEtooth:FDVTime:Y[:SCALe]:OFFSet 701
DISPlay:BLUEtooth:MEASview:DELete (No Query Form) 702
DISPlay:BLUEtooth:MEASview:NEW (No Query Form) 702
DISPlay:BLUEtooth:MEASview:SELect 703
DISPlay:BOBW:SELected:BANDwidth 704
DISPlay:CCDF:LEGend:STATe 704
DISPlay:CCDF:WINDow:TRACe:GRATicule:GRID:STATe 705
DISPlay:CONSte[:MEASview<y>]:MPHase 705
DISPlay:CONSte[:MEASview<y>]:WINDow:TRACe:GRATicule: GRID:STATe 706
DISPlay:DDEMod[:MEASview<y>]:DELete (No query form) 707
DISPlay:DDEMod:MEASview<y>[:SOURce<x>]:NEW (No query form) 708
DISPlay:DDEMod[:MEASview<y>]:SELect 708
DISPlay:DDEMod[:SOURce<x>]:RADix 709
DISPlay:DDEMod[:SOURce<x>]:X[:SCALe] 709
DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:AUTO (No query form) 710
DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:AUTO:STATe 710
DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:MAXimum? (Query only) 711
DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:MINimum? (Query only) 711
DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]: OFFSet 712
DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:OFFSet:MAXimum? (Query only) 713
DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:OFFSet:MINimum? (Query only) 713
DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:RESet (No query form) 713
DISPlay:DIAGram:X[:SCALe] 714

DISPlay:DIAGram:X[:SCALe]:RESet (No Query Form) . . . 714
DISPlay:DIQVtime[:MEASview<y>]:WINDow:TRACe:GRATicule: GRID:STATe . . . 715
DISPlay:DIQVtime[:MEASview<y>]:Y[:SCALe] . . . 715
DISPlay:DIQVtime[:MEASview<y>]:Y[:SCALe]:AUTO (No query form) . . . 716
DISPlay:DIQVtime[:MEASview<y>]:Y[:SCALe]:OFFSet . . . 716
DISPlay:DPX:DGRam:TIME[:SCALe]:OFFSet:DIVision . . . 717
DISPlay:DPX:DGRam:TIME[:SCALe]:PDIVision . . . 717
DISPlay:DPX:DGRam:TIME[:SCALe]:RESet (No Query Form) . . . 718
DISPlay:DPX:DGRam:TSTamp:STATe . . . 718
DISPlay:DPX:DGRam:Y[:SCALe]:AUTO (No Query Form) . . . 719
DISPlay:DPX:DGRam:Y[:SCALe]:RESet (No Query Form) . . . 719
DISPlay:DPX:LEGend:STATe . . . 720
DISPlay:DPX:PHASe:Y[:SCALe]:AXIS . . . 720
DISPlay:DPX:PHASe:Y[:SCALe]:AXIS:OFFSet . . . 721
DISPlay:DPX:WINDow:TRACe:GRATicule:GRID:STATe . . . 721
DISPlay:DPX:Y[:SCALe]:OFFSet . . . 722
DISPlay:DPX:Y[:SCALe]:PDIVision . . . 722
DISPlay:EDIagram[:MEASview<y>]:WINDow:TRACe:GRATicule: GRID:STATe . . . 723
DISPlay:EDIagram[:MEASview<y>]:Y[:SCALe] . . . 723
DISPlay:EDIagram[:MEASview<y>]:Y[:SCALe]:AUTO (No query form) . . . 724
DISPlay:EDIagram[:MEASview<y>]:Y[:SCALe]:OFFSet . . . 724
DISPlay:EMC:INCLude:MARGin . . . 725
DISPlay:EMC:MARKer:SHOW:STATe . . . 725
DISPlay:EMC:RESet:SCALe (No Query Form) . . . 726
DISPlay:EMC:SCALe:LOG:STATe . . . 726
DISPlay:EMC:SELect:SPOT:NUMBer . . . 727
DISPlay:EMC:SHOW:LIMit . . . 727
DISPlay:EMC:SHOW:LIMit:LEGend . . . 728
DISPlay:EMC:SHOW:SPOT:INDIcator . . . 728
DISPlay:EMC:SHOW:WIZArd . . . 729
DISPlay:EMC:SPOT<x>:SELect . . . 729
DISPlay:EMC:TRACe:LEGEnd . . . 730
DISPlay:EMC:WINDow:TRACe:GRATicule:GRID:STATe . . . 730
DISPlay:EMC:X[:SCALe]:AUTO (No Query Form) . . . 731
DISPlay:EMC:X[:SCALe]:STARt . . . 731
DISPlay:EMC:X[:SCALe]:STOP . . . 732
DISPlay:EMC:Y[:SCALe] . . . 732
DISPlay:EMC:Y[:SCALe]:AUTO (No Query Form) . . . 732
DISPlay:EMC:Y[:SCALe]:OFFSet . . . 733
DISPlay:EVM[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe . . . 733
DISPlay:EVM[:MEASview<y>]:Y[:SCALe] . . . 734
DISPlay:EVM[:MEASview<y>]:Y[:SCALe]:AUTO (No query form) . . . 735
DISPlay:EVM[:MEASview<y>]:Y[:SCALe]:OFFSet . . . 735
DISPlay:FDVTime[:MEASview<y>]:WINDow:TRACe:GRATicule: GRID:STATe . . . 736
DISPlay:FDVTime[:MEASview<y>]:Y[:SCALe] . . . 736
DISPlay:FDVTime[:MEASview<y>]:Y[:SCALe]:AUTO (No query form) . . . 737

DISPlay:FDVTime[:MEASview<y>]:Y[:SCALe]:OFFSet 738
DISPlay:{FSETtling|PSETtling}:MARKer:SHOW:STATe 738
DISPlay:{FSETtling|PSETtling}:TIME:DECimal 739
DISPlay:{FSETtling|PSETtling}:WINDow:TRACe:GRATicule: GRID:STATe 739
DISPlay:{FSETtling|PSETtling}:X[:SCALe] 740
DISPlay:{FSETtling|PSETtling}:X[:SCALe]:AUTO (No Query Form) 740
DISPlay:{FSETtling|PSETtling}:X[:SCALe]:AUTO:STATe 741
DISPlay:{FSETtling|PSETtling}:X[:SCALe]:MAXimum? (Query Only) 741
DISPlay:{FSETtling|PSETtling}:X[:SCALe]:MINimum? (Query Only) 742
DISPlay:{FSETtling|PSETtling}:X[:SCALe]:OFFSet 742
DISPlay:{FSETtling|PSETtling}:X[:SCALe]:OFFSet:MAXimum? (Query Only) 743
DISPlay:{FSETtling|PSETtling}:X[:SCALe]:OFFSet:MINimum? (Query Only) 743
DISPlay:{FSETtling|PSETtling}:Y[:SCALe] 744
DISPlay:{FSETtling|PSETtling}:Y[:SCALe]:AUTO (No Query Form) 744
DISPlay:{FSETtling|PSETtling}:Y[:SCALe]:OFFSet 745
DISPlay:{FSETtling|PSETtling}:Y[:SCALe]:PDIVision 745
DISPlay:FVTime[:MEASview<y>]:READout[:STATe] 746
DISPlay:FVTime[:MEASview<y>]:SELected:TIMestamp 746
DISPlay:FVTime[:MEASview<y>]:WINDow:TRACe:GRATicule: GRID:STATe 747
DISPlay:FVTime[:MEASview<y>]:X[:SCALe] 747
DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:AUTO (No Query Form) 748
DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:AUTO:STATe 749
DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:MAXimum? (Query Only) 749
DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:MINimum? (Query Only) 750
DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:OFFSet 750
DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MAXimum? (Query Only) 751
DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MINimum? (Query Only) 752
DISPlay:FVTime[:MEASview<y>]:Y[:SCALe] 752
DISPlay:FVTime[:MEASview<y>]:Y[:SCALe]:AUTO (No Query Form) 753
DISPlay:FVTime[:MEASview<y>]:Y[:SCALe]:OFFSet 753
DISPlay:GENeral:MEASview<y>:DELete (No Query Form) 754
DISPlay:GENeral:MEASview<y>[:SOURce<x>]:NEW (No Query Form) 755
DISPlay:GENeral:MEASview<y>:SELect 755
DISPlay:GPRF:MEASview:DELete (No Query Form) 756
DISPlay:GPRF:MEASview:NEW (No Query Form) 757
DISPlay:GPRF:MEASview:SELect 758
DISPlay:IQVTime[:MEASview<y>]:WINDow:TRACe:GRATicule: GRID:STATe 759
DISPlay:IQVTime[:MEASview<y>]:X[:SCALe] 759
DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:AUTO (No Query Form) 760
DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:AUTO:STATe 761
DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:MAXimum? (Query Only) 761
DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:MINimum? (Query Only) 762
DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:OFFSet 762
DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MAXimum? (Query Only) 763
DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MINimum? (Query Only) 764
DISPlay:IQVTime[:MEASview<y>]:Y[:SCALe] 764

DISPlay:IQVTime[:MEASview<y>]:Y[:SCALe]:AUTO (No Query Form) 765
DISPlay:IQVTime[:MEASview<y>]:Y[:SCALe]:OFFSet 765
DISPlay:IQVTime[:MEASview<y>]:Y[:SCALe]:RESCale (No Query Form) 766
DISPlay:LTE:ACLR:MARKer:SHOW:STATe 766
DISPlay:LTE:ACLR:RESet:SCALe (No Query Form) 767
DISPlay:LTE:ACLR:SHOW:LIMit 767
DISPlay:LTE:ACLR:WINDow:TRACe:GRATicule:GRID:STATe 768
DISPlay:LTE:ACLR:WINDow:TRACe:POWer:LEVEl:STATe 768
DISPlay:LTE:ACLR:X:SCALe 769
DISPLay:LTE:ACLR:X:SCALe:AUTO (No Query Form) 769
DISPlay:LTE:ACLR:X:SCALe:OFFSet 769
DISPlay:LTE:ACLR:Y:SCALe:AUTO (No Query Form) 770
DISPlay:LTE:ACLR:Y:SCALe:OFFSet 770
DISPlay:LTE:ACLR:Y:SCALe 771
DISPlay:LTE:CHSPectrum:MARKer:SHOW:STATe 771
DISPlay:LTE:CHSPectrum:RESet:SCALe (No Query Form) 772
DISPlay:LTE:CHSPectrum:WINDow:TRACe:GRATicule:GRID:STATe 772
DISPlay:LTE:CHSPectrum:X:SCALe 773
DISPLay:LTE:CHSpectrum:X:SCALe:AUTO (No Query Form) 773
DISPlay:LTE:CHSPectrum:X:SCALe:OFFSet 774
DISPlay:LTE:CHSPectrum:Y:SCALe 774
DISPlay:LTE:CHSPectrum:Y:SCALe:AUTO (No Query Form) 774
DISPlay:LTE:CHSPectrum:Y:SCALe:OFFSet 775
DISPlay:LTE:CONSte:SCALe 775
DISPlay:LTE:CONSte:X:OFFSet 776
DISPlay:LTE:CONSte:Y:OFFSet 776
DISPlay:LTE:MEASview:DELete (No Query Form) 777
DISPlay:LTE:MEASview:NEW (No Query Form) 777
DISPlay:LTE:MEASview:SELect 778
DISPlay:LTE:PVTime:MARKer:SHOW:STATe 778
DISPlay:LTE:PVTime:WINDow:TRACe:GRATicule:GRID:STATe 779
DISPlay:LTE:PVTime:X:SCALe 779
DISPLay:LTE:PVTime:X:SCALe:AUTO (No Query Form) 780
DISPlay:LTE:PVTime:X:SCALe:RESet (No Query Form) 780
DISPlay:LTE:PVTime:Y:SCALe 780
DISPlay:LTE:PVTime:Y:SCALe:AUTO (No Query Form) 781
DISPlay:LTE:PVTime:Y:SCALe:OFFSet 781
DISPlay:LTE:PVTime:Y:SCALe:PDIVision 782
DISPlay:LTE:PVTime:Y:SCALe:RESet (No Query Form) 782
DISPlay:MCPower:MARKer:SHOW:STATe 783
DISPlay:MCPower:PLEVel:SHOW:STATe 783
DISPlay:MCPower:RESet:SCALe (No Query Form) 783
DISPlay:MCPower:WINDow:TRACe:GRATicule:GRID:STATe 784
DISPlay:MCPower:X[:SCALe] 784
DISPlay:MCPower:X[:SCALe]:AUTO (No Query Form) 785
DISPlay:MCPower:X[:SCALe]:OFFSet 785

DISPlay:MCPower:Y[:SCALe] 786
DISPlay:MCPower:Y[:SCALe]:AUTO (No Query Form) 786
DISPlay:MCPower:Y[:SCALe]:OFFSet 787
DISPlay:MERRor[:MEASview<y>]:WINDow:TRACe:GRATicule: GRID:STATe 787
DISPlay:MERRor[:MEASview<y>]:Y[:SCALe] 788
DISPlay:MERRor[:MEASview<y>]:Y[:SCALe]:AUTO (No query form) 788
DISPlay:MERRor[:MEASview<y>]:Y[:SCALe]:OFFSet 789
DISPlay:NOISe:FIGure[:SCALe]:AUTO (No Query Form) 789
DISPlay:NOISe:FIGure:WINDow:TRACe:GRATicule:GRID:STATe 790
DISPlay:NOISe:FIGure:WINDow:TRACe:LEGend:STATe 790
DISPlay:NOISe:FIGure:WINDow:TRACe:MEASpoints:STATe 791
DISPlay:NOISe:FIGure:X:AUTO (No Query Form) 791
DISPlay:NOISe:FIGure:X:OFFSet 792
DISPlay:NOISe:FIGure:X[:SCALe] 792
DISPlay:NOISe:FIGure:Y:AUTO (No Query Form) 793
DISPlay:NOISe:FIGure:Y:AUTO:STATe 793
DISPlay:NOISe:FIGure:Y:PDIVision 794
DISPlay:NOISe:FIGure:Y:POSition 794
DISPlay:NOISe:FIGure:Y:SCALe 795
DISPlay:NOISe:FIGure:Y:SCALe:LINear:STATe 795
DISPlay:NOISe:GAIN[:SCALe]:AUTO (No Query Form) 796
DISPlay:NOISe:GAIN:WINDow:TRACe:GRATicule:GRID:STATe 796
DISPlay:NOISe:GAIN:WINDow:TRACe:LEGend:STATe 797
DISPlay:NOISe:GAIN:WINDow:TRACe:MEASPoints:STATe 797
DISPlay:NOISe:GAIN:X:AUTO (No Query Form) 798
DISPlay:NOISe:GAIN:X:OFFSet 798
DISPlay:NOISe:GAIN:X:SCALe 798
DISPlay:NOISe:GAIN:Y:AUTO (No Query Form) 799
DISPlay:NOISe:GAIN:Y:AUTO:STATe 799
DISPlay:NOISe:GAIN:Y:PDIVision 800
DISPlay:NOISe:GAIN:Y:POSition 801
DISPlay:NOISe:GAIN:Y:SCALe 801
DISPlay:NOISe:MEASview:DELete (No Query Form) 802
DISPlay:NOISe:MEASview:NEW (No Query Form) 802
DISPlay:NOISe:MEASview:SELect 803
DISPlay:NOISe:POWer:LINear:STATe 804
DISPlay:NOISe:TEMPerature[:SCALe]:AUTO (No Query Form) 804
DISPlay:NOISe:TEMPerature:WINDow:TRACe:GRATicule:GRID:STATe 804
DISPlay:NOISe:TEMPerature:WINDow:TRACe:LEGend:STATe 805
DISPlay:NOISe:TEMPerature:WINDow:TRACe:MEASPoints:STATe 806
DISPlay:NOISe:TEMPerature:X:AUTO (No Query Form) 806
DISPlay:NOISe:TEMPerature:X:OFFSet 807
DISPlay:NOISe:TEMPerature:X[:SCALe] 807
DISPlay:NOISe:TEMPerature:X:STARt 808
DISPlay:NOISe:TEMPerature:X:STOP 808
DISPlay:NOISe:TEMPerature:Y:AUTO (No Query Form) 809

DISPlay:NOISe:TEMPerature:Y:AUTO:STATe 809
DISPlay:NOISe:TEMPerature:Y:PDIVision 809
DISPlay:NOISe:TEMPerature:Y:POSition 810
DISPlay:NOISe:TEMPerature:Y:SCALe 811
DISPlay:NOISe:UNCERTainty:RESult:GAIN? (Query Only) 811
DISPlay:NOISe:UNCERTainty:RESult:NFIGure? (Query Only) 811
DISPlay:NOISe:YFACtor[:SCALe]:AUTO (No Query Form) 812
DISPlay:NOISe:YFACtor:WINDow:TRACe:GRATicule:GRID:STATe 812
DISPlay:NOISe:YFACtor:WINDow:TRACe:LEGend:STATe 813
DISPlay:NOISe:YFACtor:WINDow:TRACe:MEASPoints:STATe 813
DISPlay:NOISe:YFACtor:X:AUTO (No Query Form) 814
DISPlay:NOISe:YFACtor:X:OFFSet 814
DISPlay:NOISe:YFACtor:X[:SCALe] 815
DISPlay:NOISe:YFACtor:X:STARt 815
DISPlay:NOISe:YFACtor:X:STOP 816
DISPlay:NOISe:YFACtor:Y:AUTO (No Query Form) 816
DISPlay:NOISe:YFACtor:Y:AUTO:STATe 817
DISPlay:NOISe:YFACtor:Y:PDIVision 817
DISPlay:NOISe:YFACtor:Y:POSition 818
DISPlay:NOISe:YFACtor:Y:SCALe 818
DISPlay:OBWidth:MARKer:SHOW:STATe 819
DISPlay:OBWidth:RESet:SCALe (No Query Form) 819
DISPlay:OBWidth:SELected:BANDwidth 820
DISPlay:OBWidth:WINDow:TRACe:GRATicule:GRID:STATe 820
DISPlay:OBWidth:X[:SCALe] 820
DISPlay:OBWidth:X[:SCALe]:AUTO (No Query Form) 821
DISPlay:OBWidth:X[:SCALe]:OFFSet 821
DISPlay:OBWidth:Y[:SCALe] 822
DISPlay:OBWidth:Y[:SCALe]:AUTO (No Query Form) 822
DISPlay:OBWidth:Y[:SCALe]:OFFSet 823
DISPlay:OFDM:CONSte[:SCALe] 823
DISPlay:OFDM:CONSte:X:OFFSet 824
DISPlay:OFDM:CONSte:Y:OFFSet 824
DISPlay:OFDM:CRESponse:FREQuency:AUTO (No Query Form) 824
DISPlay:OFDM:CRESponse:FREQuency:OFFSet 825
DISPlay:OFDM:CRESponse:FREQuency[:SCALe] 825
DISPlay:OFDM:CRESponse:MAGNitude:AUTO (No Query Form) 826
DISPlay:OFDM:CRESponse:MAGNitude:OFFSet 826
DISPlay:OFDM:CRESponse:MAGNitude:PDIVision 826
DISPlay:OFDM:CRESponse:PHASe:AUTO (No Query Form) 827
DISPlay:OFDM:CRESponse:PHASe:OFFSet 827
DISPlay:OFDM:CRESponse:PHASe[:SCALe] 828
DISPlay:OFDM:CRESponse:WINDow:SELect:PLOT 828
DISPlay:OFDM:CRESponse:WINDow:TRACe:GRATicule:GRID:STATe 828
DISPlay:OFDM:EVM:FREQuency:AUTO (No Query Form) 829
DISPlay:OFDM:EVM:FREQuency:OFFSet 829

DISPlay:OFDM:EVM:FREQuency[:SCALe] 830
DISPlay:OFDM:EVM:MARKer:SHOW:STATe 830
DISPlay:OFDM:EVM:TIME:AUTO (No Query Form) 830
DISPlay:OFDM:EVM:TIME:OFFSet 831
DISPlay:OFDM:EVM:TIME[:SCALe] 831
DISPlay:OFDM:EVM:WINDow:SELect:PLOT 832
DISPlay:OFDM:EVM:WINDow:TRACe:GRATicule:GRID:STATe 832
DISPlay:OFDM:EVM:Y:AUTO (No Query Form) 832
DISPlay:OFDM:EVM:Y:OFFSet 833
DISPlay:OFDM:EVM:Y[:SCALe] 833
DISPlay:OFDM:FLATness:AUTO (No Query Form) 834
DISPlay:OFDM:FLATness:X:AUTO (No Query Form) 834
DISPlay:OFDM:FLATness:X:OFFSet 834
DISPlay:OFDM:FLATness:X[:SCALe] 835
DISPlay:OFDM:FLATness:Y:AUTO (No Query Form) 835
DISPlay:OFDM:FLATness:Y:OFFSet 836
DISPlay:OFDM:FLATness:Y[:SCALe] 836
DISPlay:OFDM:MEASview:DELete (No Query Form) 837
DISPlay:OFDM:MEASview:NEW (No Query Form) 837
DISPlay:OFDM:MEASview:SELect 838
DISPlay:OFDM:MERRor:FREQuency[:SCALe] 839
DISPlay:OFDM:MERRor:FREQuency[:SCALe]:AUTO (No Query Form) 839
DISPlay:OFDM:MERRor:FREQuency[:SCALe]:OFFSet 839
DISPlay:OFDM:MERRor:MARKer:SHOW:STATe 840
DISPlay:OFDM:MERRor:TIME[:SCALe] 840
DISPlay:OFDM:MERRor:TIME[:SCALe]:AUTO (No Query Form) 841
DISPlay:OFDM:MERRor:TIME[:SCALe]:OFFSet 841
DISPlay:OFDM:MERRor:WINDow:SELect:PLOT 841
DISPlay:OFDM:MERRor:WINDow:TRACe:GRATicule:GRID:STATe 842
DISPlay:OFDM:MERRor:Y:AUTO (No Query Form) 842
DISPlay:OFDM:MERRor:Y:OFFSet 843
DISPlay:OFDM:MERRor:Y[:SCALe] 843
DISPlay:OFDM:PERRor:FREQuency[:SCALe] 843
DISPlay:OFDM:PERRor:FREQuency[:SCALe]:AUTO (No Query Form) 844
DISPlay:OFDM:PERRor:FREQuency[:SCALe]:OFFSet 844
DISPlay:OFDM:PERRor:MARKer:SHOW:STATe 845
DISPlay:OFDM:PERRor:TIME[:SCALe] 845
DISPlay:OFDM:PERRor:TIME[:SCALe]:AUTO (No Query Form) 845
DISPlay:OFDM:PERRor:TIME[:SCALe]:OFFSet 846
DISPlay:OFDM:PERRor:WINDow:SELect:PLOT 846
DISPlay:OFDM:PERRor:WINDow:TRACe:GRATicule:GRID:STATe 847
DISPlay:OFDM:PERRor:Y:AUTO (No Query Form) 847
DISPlay:OFDM:PERRor:Y:OFFSet 847
DISPlay:OFDM:PERRor:Y[:SCALe] 848
DISPlay:OFDM:POWer:FREQuency[:AUTO] (No Query Form) 848
DISPlay:OFDM:POWer:FREQuency[:OFFSet] 849

DISPlay:OFDM:POWer:FREQuency[:SCALe] 849
DISPlay:OFDM:POWer:MARKer:SHOW:STATe 849
DISPlay:OFDM:POWer:TIME[:AUTO] (No Query Form) 850
DISPlay:OFDM:POWer:TIME:OFFSet 850
DISPlay:OFDM:POWer:TIME[:SCALe] 851
DISPlay:OFDM:POWer:WINDow:SELect:PLOT 851
DISPlay:OFDM:POWer:WINDow:TRACe:GRATicule:GRID:STATe 851
DISPlay:OFDM:POWer:Y:AUTO (No Query Form) 852
DISPlay:OFDM:POWer:Y:OFFSet 852
DISPlay:OFDM:POWer:Y[:SCALe] 853
DISPlay:P25:CONSte:WINDow:TRACe:GRATicule:GRID:STATe 853
DISPlay:P25:EDIagram:WINDow:TRACe:GRATicule:GRID:STATe 853
DISPlay:P25:EDIagram:Y[:SCALe] 854
DISPlay:P25:EDIagram:Y[:SCALe]:AUTO (No Query Form) 855
DISPlay:P25:EDIagram:Y[:SCALe]:OFFSet 855
DISPlay:P25:MEASview:DELete (No Query Form) 855
DISPlay:P25:MEASview:NEW (No Query Form) 856
DISPlay:P25:MEASview:SELect 857
DISPlay:P25:PVTime:BURSt:X[:SCALe] 857
DISPlay:P25:PVTime:BURSt:X[:SCALe]:AUTO (No Query Form) 858
DISPlay:P25:PVTime:BURSt:X[:SCALe]:OFFSet 858
DISPlay:P25:PVTime:BURSt:X[:SCALe]:RESet (No Query Form) 859
DISPlay:P25:PVTime:FALL:X[:SCALe] 859
DISPlay:P25:PVTime:FALL:X[:SCALe]:AUTO (No Query Form) 860
DISPlay:P25:PVTime:FALL:X[:SCALe]:OFFSet 860
DISPlay:P25:PVTime:FALL:X[:SCALe]:RESet (No Query Form) 861
DISPlay:P25:PVTime:MARKer:SHOW:STATe 861
DISPlay:P25:PVTime:RISE:X[:SCALe] 862
DISPlay:P25:PVTime:RISE:X[:SCALe]:AUTO (No Query Form) 862
DISPlay:P25:PVTime:RISE:X[:SCALe]:OFFSet 863
DISPlay:P25:PVTime:RISE:X[:SCALe]:RESet (No Query Form) 863
DISPlay:P25:PVTime:WINDow:SELect:PLOT 864
DISPlay:P25:PVTime:WINDow:TRACe:GRATicule:GRID:STATe 864
DISPlay:P25:PVTime:Y[:SCALe] 865
DISPlay:P25:PVTime:Y[:SCALe]:AUTO (No Query Form) 865
DISPlay:P25:PVTime:Y[:SCALe]:OFFSet 866
DISPlay:P25:PVTime:Y[:SCALe]:PDIVision 866
DISPlay:P25:RADix 867
DISPlay:PERRor[:MEASview<y>]:WINDow:TRACe:GRATicule: GRID:STATe 867
DISPlay:PERRor[:MEASview<y>]:Y[:SCALe] 868
DISPlay:PERRor[:MEASview<y>]:Y[:SCALe]:AUTO (No query form) 869
DISPlay:PERRor[:MEASview<y>]:Y[:SCALe]:OFFSet 869
DISPlay:PHVTime[:MEASview<y>]:SELected:TIMestamp 870
DISPlay:PHVTime[:MEASview<y>]:WINDow:TRACe:GRATicule: GRID:STATe 870
DISPlay:PHVTime[:MEASview<y>]:X[:SCALe] 871
DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:AUTO (No Query Form) 871

DISPlay:PHVTime[:MEASview?]:Y[:SCALe]:AXIS:ENABle:REFerence {set and query} 872
DISPlay:PHVTime[:MEASview<y>]:READout[:STATe] 872
DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:AUTO:STATe 873
DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:MAXimum? (Query Only) 873
DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:MINimum? (Query Only) 874
DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:OFFSet 875
DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MAXimum? (Query Only) 875
DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MINimum? (Query Only) 876
DISPlay:PHVTime[:MEASview<y>]:Y[:SCALe] 876
DISPlay:PHVTime[:MEASview<y>]:Y[:SCALe]:AUTO (No Query Form) 877
DISPlay:PHVTime[:MEASview<y>]:Y[:SCALe]:AXIS 877
DISPlay:PHVTime[:MEASview<y>]:Y[:SCALe]:AXIS:REFerence 878
DISPlay:PHVTime[:MEASview<y>]:Y[:SCALe]:OFFSet 878
DISPlay:PHVTime[:MEASview<y>]:Y[:SCALe]:RESCale (No Query Form) 879
DISPlay:PNOise:LEGend:STATe 879
DISPlay:PNOise:MARKer:SHOW:STATe 880
DISPlay:PNOise:RESet:SCALe (No Query Form) 880
DISPlay:PNOise:WINDow:TRACe:GRATicule:GRID:STATe 880
DISPlay:PNOise:X[:SCALe]:AUTO (No Query Form) 881
DISPlay:PNOise:X[:SCALe]:STARt 881
DISPlay:PNOise:X[:SCALe]:STOP 882
DISPlay:PNOise:Y[:SCALe] 882
DISPlay:PNOise:Y[:SCALe]:AUTO (No Query Form) 882
DISPlay:PNOise:Y[:SCALe]:OFFSet 883
DISPlay:PNOise:Y[:SCALe]:PDIVision 883
DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]:CURRent:BIN 884
DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]:SCALe:AUTO (No Query Form) 884
DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]:SCALe:RESEt (No Query Form) 885
DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]: SCALe:X:AUTO (No Query Form) 885
DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]: SCALe:X:NUMBer 886
DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]: SCALe:X:OFFSet 886
DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]: SCALe:X:RSCale (No Query Form) 887
DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]: SCALe:Y:AUTO (No Query Form) 887
DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]:SCALe:Y:FULL 888
DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]: SCALe:Y:OFFSet 888
DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]: SCALe:Y:RSCale (No Query Form) 889
DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]: SCALe:Y:STOP? (Query Only) 889
DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]:WINDow: TRACe:GRATicule:GRID:STATe 890
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:MFReqerror 890
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:ATX 891
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:AVERage 891
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:DRODb 892
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:DROop 892
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:DUTPct 893
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:DUTRatio 893
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:FABS 894

DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:FALL . . . 894
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:FDELta . . . 895
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:FRDeviation . . . 895
DISPLAY:PULSe:CUMulative:STATistics[:MEASview<y>]:IRAMplitude . . . 896
DISPLAY:PULSe:CUMulative:STATistics[:MEASview<y>]:IRTime . . . 897
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:MPHerror . . . 897
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:OVEDb . . . 898
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:OVERshoot . . . 898
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:PHDeviation . . . 899
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:PPFD . . . 899
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:PPFRequency . . . 900
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:PPOWer . . . 900
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:PPPD . . . 901
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:PPPHase . . . 901
DISPLAY:PULSe:CUMulative:STATistics[:MEASview<y>]:RINTerval . . . 902
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:RIPDb . . . 902
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:RIPPle . . . 903
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:RISE . . . 904
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:RMSFreqerror . . . 904
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:MFReqerror . . . 905
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:RMSPherror . . . 905
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:RRATe . . . 906
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:WIDTh . . . 906
DISPlay:PULSe:MEASview<y>:DELete (No Query Form) . . . 907
DISPlay:PULSe:MEASview<y>[:SOURce<x>]:NEW (No Query Form) . . . 907
DISPlay:PULSe:MEASview<y>:SELect . . . 908
DISPlay:PULSe:OGRAm[:MEASview<y>] . . . 909
DISPlay:PULSe:OGRAm[:MEASview<y>]:FRAME:INFOrmation . . . 909
DISPlay:PULSe:OGRAm[:MEASview<y>]:MARKer:FRAME . . . 910
DISPlay:PULSe:OGRAm[:MEASview<y>]:MARKer:SHOW:STATe . . . 910
DISPlay:PULSe:OGRAm[:MEASview<y>][:SCALe]:X:FVTime:FREQuency:STARt . . . 911
DISPlay:PULSe:OGRAm[:MEASview<y>][:SCALe]:X:FVTime:FREQuency:STOP . . . 911
DISPlay:PULSe:OGRAm[:MEASview<y>][:SCALe]:X:FVTime:RSCale (No Query Form) . . . 912
DISPlay:PULSe:OGRAm[:MEASview<y>][:SCALe]:X:RSCale (No Query Form) . . . 912
DISPlay:PULSe:OGRAm[:MEASview<y>][:SCALe]:X:TVTime (No Query Form) . . . 913
DISPlay:PULSe:OGRAm[:MEASview<y>][:SCALe]:X:TVTime:OFFSet (No Query Form) . . . 913
DISPlay:PULSe:OGRAm[:MEASview<y>][:SCALe]:X:TVTime:RSCale (No Query Form) . . . 914
DISPlay:PULSe:OGRAm[:MEASview<y>]:FVTime:DETection . . . 914
DISPlay:PULSe:OGRAm[:MEASview<y>][:SCALe]:RSCale . . . 915
DISPlay:PULSe:OGRAm[:MEASview<y>]:WINDow:TRACe: GRATicule:GRID:STATe . . . 915
DISPlay:PULSe:OGRAm[:MEASview<y>]:TRACE:POINts . . . 916
DISPlay:PULSe:OGRAm[:MEASview<y>][:SCALe]:Y:FRAMe:FULL . . . 916
DISPlay:PULSe:OGRAm:SCALe:Y:FRAMe:OFFSet . . . 917
DISPlay:PULSe:OGRAm[:MEASview<y>]:TVTime:SMOOth:POINts . . . 917
DISPlay:PULSe:OGRAm[:MEASview<y>] . . . 918
DISPlay:PULSe:RESult[:MEASview<y>]:ATX . . . 918

DISPlay:PULSe:RESult[:MEASview<y>]:AVERage 919
DISPlay:PULSe:RESult[:MEASview<y>]:DRODb 919
DISPlay:PULSe:RESult[:MEASview<y>]:DROop 920
DISPlay:PULSe:RESult[:MEASview<y>]:DUTPct 920
DISPlay:PULSe:RESult[:MEASview<y>]:DUTRatio 921
DISPlay:PULSe:RESult[:MEASview<y>]:FABS 921
DISPlay:PULSe:RESult[:MEASview<y>]:FALL 922
DISPlay:PULSe:RESult[:MEASview<y>]:FDELta 922
DISPlay:PULSe:RESult[:MEASview<y>]:FRDeviation 923
DISPlay:PULSe:RESult[:MEASview<y>]:IRAMplitude 923
DISPlay:PULSe:RESult[:MEASview<y>]:IRTime 924
DISPlay:PULSe:RESult[:MEASview<y>]:MFREqerror 924
DISPlay:PULSe:RESult[:MEASview<y>]:MPHerror 925
DISPlay:PULSe:RESult[:MEASview<y>]:OVEDb 925
DISPlay:PULSe:RESult[:MEASview<y>]:OVERshoot 926
DISPlay:PULSe:RESult[:MEASview<y>]:PHDeviation 926
DISPlay:PULSe:RESult[:MEASview<y>]:PPPD 927
DISPlay:PULSe:RESult[:MEASview<y>]:PPFD 927
DISPlay:PULSe:RESult[:MEASview<y>]:PPFRequency 928
DISPlay:PULSe:RESult[:MEASview<y>]:PPOWer 928
DISPlay:PULSe:RESult[:MEASview<y>]:PPPHase 929
DISPlay:PULSe:RESult[:MEASview<y>]:RINTerval 929
DISPlay:PULSe:RESult[:MEASview<y>]:RIPDb 930
DISPlay:PULSe:RESult:MEASview<y>:RIPPle 930
DISPlay:PULSe:RESult[:MEASview<y>]:RISE 931
DISPlay:PULSe:RESult[:MEASview<y>]:RMSFreqerror 931
DISPlay:PULSe:RESult[:MEASview<y>]:RMSPherror 932
DISPlay:PULSe:RESult[:MEASview<y>]:RRATe 932
DISPlay:PULSe:RESult[:MEASview<y>]:TIME 933
DISPlay:PULSe:RESult[:MEASview<y>]:WIDTh 933
DISPlay:PULSe[:MEASview<y>]:SELect:NUMBer 934
DISPlay:PULSe[:MEASview<y>]:SELect:RESult 934
DISPlay:PULSe[:MEASview<y>]:SELect:RESult 936
DISPlay:PULSe:STATistics[:MEASview<y>]:MARKer<x>:SHOW:STATe 937
DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT 938
DISPlay:PULSe:STATistics[:MEASview<y>]:WINDow:TRACe: GRATicule:GRID:STATe 938
DISPlay:PULSe:STATistics[:MEASview<y>]:X:RSCale (No Query Form) 939
DISPlay:PULSe:STATistics[:MEASview<y>]:X[:SCALe]:NUMBer 939
DISPlay:PULSe:STATistics[:MEASview<y>]:X:SCALe:OFFSet 940
DISPlay:PULSe:STATistics[:MEASview<y>]:Y:RSCale (No Query Form) 941
DISPlay:PULSe:STATistics[:MEASview<y>]:Y[:SCALe]:FULL 941
DISPlay:PULSe:STATistics[:MEASview<y>]:Y[:SCALe]:OFFSet 942
DISPlay:PULSe:STATistics[:MEASview<y>]:Y[:SCALe]:STOP? (Query Only) 943
DISPlay:PULSe:TRACe[:MEASview<y>]:MARKer:SHOW:STATe 944
DISPlay:PULSe:TRACe[:MEASview<y>]:POINt:SHOW 944
DISPlay:PULSe:TRACe[:MEASview<y>]:WINDow:TRACe: GRATicule:GRID:STATe 945

DISPlay:PULSe:TRACe[:MEASview<y>]:X:RSCale 945
DISPlay:PULSe:TRACe[:MEASview<y>]:X[:SCALe] 946
DISPlay:PULSe:TRACe[:MEASview<y>]:X[:SCALe]:FULL 946
DISPlay:PULSe:TRACe[:MEASview<y>]:X[:SCALe]:OFFSet 947
DISPlay:PULSe:TRACe[:MEASview<y>]:X[:SCALe]:PDIVision 947
DISPlay:PULSe:TRACe[:MEASview<y>]:Y:RSCale 948
DISPlay:PULSe:TRACe[:MEASview<y>]:Y[:SCALe]:FULL 948
DISPlay:PULSe:TRACe[:MEASview<y>]:Y[:SCALe]:OFFSet 949
DISPlay:PULSe:TRACe[:MEASview<y>]:Y[:SCALe]:STOP 949
DISPlay:SEM:MARKer:SHOW:STATe 950
DISPlay:SEM:RESet:SCALe (No Query Form) 950
DISPlay:SEM:SHOW:LIMit 951
DISPlay:SEM:WINDow:TRACe:GRATicule:GRID:STATe 951
DISPlay:SEM:X[:SCALe]:AUTO (No Query Form) 952
DISPlay:SEM:X[:SCALe]:START 952
DISPlay:SEM:X[:SCALe]:STOP 952
DISPlay:SEM:Y[:SCALe] 953
DISPlay:SEM:Y[:SCALe]:AUTO (No Query Form) 953
DISPlay:SEM:Y[:SCALe]:OFFSet 954
DISPlay:SGRam[:MEASview<y>]:FREQuency:AUTO (No Query Form) 954
DISPlay:SGRam[:MEASview<y>]:FREQuency:OFFSet 954
DISPlay:SGRam[:MEASview<y>]:FREQuency[:SCAle] 955
DISPlay:SGRam[:MEASview<y>]:MARKer:SHOW:STATe 956
DISPlay:SGRam[:MEASview<y>]:SELected:TIMestamp 956
DISPlay:SGRam[:MEASview<y>]:TIME:AUTO (No Query Form) 957
DISPlay:SGRam[:MEASview<y>]:TIME:OFFSet 957
DISPlay:SGRam[:MEASview<y>]:TIME:OFFSet:DIVisions 958
DISPlay:SGRam[:MEASview<y>]:TIME:OVERlap 958
DISPlay:SGRam[:MEASview<y>]:TIME:OVERlap:PERCent 959
DISPlay:SGRam[:MEASview<y>]:TIME[:SCAle] 959
DISPlay:SGRam[:MEASview<y>]:TIME[:SCAle]:PER:DIVision 960
DISPlay:SGRam[:MEASview<y>]:TIME:SPECtrums:PERLine (Query Only) 961
DISPlay:SPECtrum[:MEASview<y>]:FREQuency:AUTO (No Query Form) 961
DISPlay:SPECtrum[:MEASview<y>]:FREQuency:OFFSet 961
DISPlay:SPECtrum[:MEASview<y>]:FREQuency[:SCAle] 962
DISPlay:SPECtrum[:MEASview<y>]:MARKer:NOISe:MODE 963
DISPlay:SPECtrum[:MEASview<y>]:READout[:STATe] 963
DISPlay:SPECtrum[:MEASview<y>][:SCALe]:LOG:STATe 964
DISPlay:SPECtrum[:MEASview<y>]:WINDow:TRACe: GRATicule:GRID:STATe 964
DISPlay:SPECtrum[:MEASview<y>]:WINDow:TRACe:LEGend:STATe 965
DISPlay:SPECtrum[:MEASview<y>]:X:LABel 965
DISPlay:SPECtrum[:MEASview<y>]:Y[:SCAle] 966
DISPlay:SPECtrum[:MEASview<y>]:Y[:SCAle]:AUTO (No Query Form) 966
DISPlay:SPECtrum[:MEASview<y>]:Y[:SCAle]:OFFSET 967
DISPlay:SPECtrum[:MEASview<y>]:Y[:SCAle]:PDIVision 967
DISPlay:SPECtrum[:MEASview<y>]:Y[:SCAle]:RESet (No Query Form) 968

DISPlay:SPURious:MARKer:SHOW:STATe 968
DISPlay:SPURious:RESet:SCALe (No Query Form) 969
DISPlay:SPURious:SCALe:LOG:STATe 969
DISPlay:SPURious:SELect:NUMBer 970
DISPlay:SPURious:SHOW:LIMit 970
DISPlay:SPURious:WINDow:TRACe:GRATicule:GRID:STATe 970
DISPlay:SPURious:X[:SCALe]:AUTO (No Query Form) 971
DISPlay:SPURious:X[:SCALe]:STARt 971
DISPlay:SPURious:X[:SCALe]:STOP 972
DISPlay:SPURious:Y[:SCALe] 972
DISPlay:SPURious:Y[:SCALe]:AUTO (No Query Form) 973
DISPlay:SPURious:Y[:SCALe]:OFFSet 973
DISPlay:TDIagram[:MEASview<y>]:WINDow:TRACe:GRATicule: GRID:STATe 974
DISPlay:TDIagram[:MEASview<y>]:Y[:SCALe] 974
DISPlay:TDIagram[:MEASview<y>]:Y[:SCALe]:AUTO (No query form) 975
DISPlay:TDIagram[:MEASview<y>]:Y[:SCALe]:OFFSet 975
DISPlay:TG:MEASview:DELete TXGain (No Query Form) 976
DISPlay:TG:MEASview:NEW TXGain (No Query Form) 976
DISPlay:TG:MEASview:SELect TXGain 976
DISPlay:TOVerview[:MEASview<y>]:WINDow:NAVigator:STATe 977
DISPlay:TOVerview[:MEASview<y>]:WINDow:TIME:MODE 978
DISPlay:TOVerview[:MEASview<y>]:WINDow:TRACe:GRATicule: GRID:STATe 978
DISPlay:TOVerview[:MEASview<y>]:WINDow:TRACe:LEGend:STATe 979
DISPlay:TOVerview[:MEASview<y>]:X[:SCALe] 979
DISPlay:TOVerview[:MEASview<y>]:X[:SCALe]:AUTO (No Query Form) 980
DISPlay:TOVerview[:MEASview<y>]:X[:SCALe]:OFFSet 980
DISPlay:TOVerview[:MEASview<y>]:Y[:SCALe] 981
DISPlay:TOVerview[:MEASview<y>]:Y[:SCALe]:AUTO (No Query Form) 981
DISPlay:TOVerview[:MEASview<y>]:Y[:SCALe]:OFFSet 982
DISPlay:TOVerview[:MEASview<y>]:Y[:SCALe]:RESCale (No Query Form) 982
DISPlay:TXGain:MARKer:SHOW:STATe 983
DISPlay:TXGain[:SCALe]:AUTO (No Query Form) 983
DISPlay:TXGain:WINDow:TRACe:GRATicule:GRID:STATe 984
DISPlay:TXGain:WINDow:TRACe:POINts:SHOW:STATe 984
DISPlay:TXGain:X[:SCALe]:AUTO (No Query Form) 985
DISPlay:TXGain:X[:SCALe]:LOG:STATe 985
DISPlay:TXGain:X[:SCALe]:STARt 985
DISPlay:TXGain:X[:SCALe]:STOP 986
DISPlay:TXGain:Y[:SCALe] 986
DISPlay:TXGain:Y[:SCALe]:AUTO (No Query Form) 987
DISPlay:TXGain:Y[:SCALe]:BOTTom 987
DISPlay:TXGain:Y[:SCALe]:PDIVision 988
DISPlay:TXGain:Y[:SCALe]:POSition 988
DISPlay:TXGain:Y[:SCALe]:TOP 989
:DISPlay:WINDow:ACQuisition:BANdwidth:SHOw:STATe {set and query} 989
DISPlay:WINDow:ACTive:MEASurement? (Query Only) 989

DISPlay:WINDow:APPLy:SOURce:COLOr {set and query} 991
DISPlay:WINDow:COLor:SCHeme 992
DISPlay:WINDow:OPTimized:MEASurement? (Query Only) 992
DISPlay:WLAN:AD:CONSte:WINDow:TRACe:GRATicule:GRID:STATe 993
DISPlay:WLAN:AD:CONSte:WINDow:TRACe:MARKer:STATe 994
DISPlay:WLAN:AD:CONSte:ZOOM:SCALe 994
DISPlay:WLAN:AD:CONSte:ZOOM[:SCALe]:X:OFFSet 995
DISPlay:WLAN:AD:CONSte:ZOOM[:SCALe]:Y:OFFSet 995
DISPlay:WLAN:AD:DATA:RADix 996
DISPlay:WLAN:AD:EVM:SCALe:AUTO (No Query Form) 996
DISPlay:WLAN:AD:EVM:WINDow:TRACe:GRATicule:GRID:STATe 997
DISPlay:WLAN:AD:EVM:WINDow:TRACe:MARKer:STATe 997
DISPlay:WLAN:AD:EVM:X:SCALe 998
DISPlay:WLAN:AD:EVM:X:SCALe:AUTO (No Query Form) 998
DISPlay:WLAN:AD:EVM:X:SCALe:OFFSet 999
DISPlay:WLAN:AD:EVM:Y:SCALe 999
DISPlay:WLAN:AD:EVM:Y:SCALe:AUTO (No Query Form) 999
DISPlay:WLAN:AD:EVM:Y:SCALe:OFFSet 1000
DISPlay:WLAN:AD:MEASview:DELete (No Query Form) 1000
DISPlay:WLAN:AD:MEASview:NEW (No Query Form) 1001
DISPlay:WLAN:AD:MEASview:SELect 1001
DISPlay:WLAN:AY:CONSte:WINDow:TRACe:GRATicule:GRID:STATe 1002
DISPlay:WLAN:AY:CONSte:WINDow:TRACe:MARKer:STATe 1002
DISPlay:WLAN:AY:CONSte:ZOOM:SCALe 1003
DISPlay:WLAN:AY:CONSte:ZOOM:[SCALe]:X:OFFSet 1004
DISPlay:WLAN:AY:CONSte:ZOOM:[SCALe]:Y:OFFSet 1004
DISPlay:WLAN:AY:DATA:RADix 1005
DISPlay:WLAN:AY:EVM:SCALe:AUTO (No Query Form) 1005
DISPlay:WLAN:AY:EVM:WINDow:TRACe:GRATicule:GRID:STATe 1005
DISPlay:WLAN:AY:EVM:WINDow:TRACe:MARKer:STATe 1006
DISPlay:WLAN:AY:EVM:X:SCALe 1007
DISPlay:WLAN:AY:EVM:X:SCALe:AUTO (No Query Form) 1007
DISPlay:WLAN:AY:EVM:X:SCALe:OFFSet 1007
DISPlay:WLAN:AY:EVM:Y:SCALe 1008
DISPlay:WLAN:AY:EVM:Y:SCALe:AUTO (No Query Form) 1008
DISPlay:WLAN:AY:EVM:Y:SCALe:OFFSet 1009
DISPlay:WLAN:AY:MEASview:DELete (No Query Form) 1009
DISPlay:WLAN:AY:MEASview:NEW (No Query Form) 1010
DISPlay:WLAN:AY:MEASview:SELect 1010
DISPlay:WLAN:CONSte[:SCALe] 1011
DISPlay:WLAN:CONSte:X:OFFSet 1012
DISPlay:WLAN:CONSte:Y:OFFSet 1012
DISPlay:WLAN:CRESponse:AUTO (No Query Form) 1013
DISPlay:WLAN:CRESponse:FREQuency:AUTO (No Query Form) 1013
DISPlay:WLAN:CRESponse:FREQuency:STARt 1014
DISPlay:WLAN:CRESponse:FREQuency:STOP 1014

DISPlay:WLAN:CRESponse:MAGNitude:AUTO (No Query Form) 1015
DISPlay:WLAN:CRESponse:MAGNitude:OFFSet 1015
DISPlay:WLAN:CRESponse:MAGNitude[:SCALe] 1015
DISPlay:WLAN:CRESponse:MARKer:SHOW:STATe 1016
DISPlay:WLAN:CRESponse:PHASe:AUTO (No Query Form) 1016
DISPlay:WLAN:CRESponse:PHASe:OFFSet 1017
DISPlay:WLAN:CRESponse:PHASe[:SCALe] 1017
DISPlay:WLAN:CRESponse:WINDow:SELect:PLOT 1018
DISPlay:WLAN:CRESponse:WINDow:TRACe:GRATicule:GRID:STATe 1018
DISPlay:WLAN:EVM:FREQuency:AUTO (No Query Form) 1019
DISPlay:WLAN:EVM:FREQuency:RESet (No Query Form) 1019
DISPlay:WLAN:EVM:FREQuency:STARt 1019
DISPlay:WLAN:EVM:FREQuency:STOP 1020
DISPlay:WLAN:EVM:MARKer:SHOW:STATe 1020
DISPlay:WLAN:EVM[:SCALe]:AUTO (No Query Form) 1021
DISPlay:WLAN:EVM:TIME:AUTO (No Query Form) 1021
DISPlay:WLAN:EVM:TIME:AUTO:STATe 1022
DISPlay:WLAN:EVM:TIME:RESet (No Query Form) 1022
DISPlay:WLAN:EVM:TIME:STARt 1023
DISPlay:WLAN:EVM:TIME:STOP 1023
DISPlay:WLAN:EVM:WINDow:SELect:PLOT 1023
DISPlay:WLAN:EVM:WINDow:TRACe:GRATicule:GRID:STATe 1024
DISPlay:WLAN:EVM:Y:AUTO (No Query Form) 1024
DISPlay:WLAN:EVM:Y:OFFSet 1025
DISPlay:WLAN:EVM:Y:RESet (No Query Form) 1025
DISPlay:WLAN:EVM:Y[:SCALe] 1026
DISPlay:WLAN:FLATness:AUTO (No Query Form) 1026
DISPlay:WLAN:FLATness:X:AUTO (No Query Form) 1027
DISPlay:WLAN:FLATness:X:OFFSet 1027
DISPlay:WLAN:FLATness:X[:SCALe] 1028
DISPlay:WLAN:FLATness:Y:AUTO (No Query Form) 1028
DISPlay:WLAN:FLATness:Y:OFFSet 1029
DISPlay:WLAN:FLATness:Y[:SCALe] 1029
DISPlay:WLAN:MEASview:DELete (No Query Form) 1030
DISPlay:WLAN:MEASview:NEW (No Query Form) 1030
DISPlay:WLAN:MEASview:SELect 1031
DISPlay:WLAN:MERRor:FREQuency:AUTO (No Query Form) 1032
DISPlay:WLAN:MERRor:FREQuency:RESet (No Query Form) 1032
DISPlay:WLAN:MERRor:FREQuency:STARt 1033
DISPlay:WLAN:MERRor:FREQuency:STOP 1033
DISPlay:WLAN:MERRor:MARKer:SHOW:STATe 1034
DISPlay:WLAN:MERRor[:SCALe]:AUTO (No Query Form) 1034
DISPlay:WLAN:MERRor:TIME:AUTO (No Query Form) 1035
DISPlay:WLAN:MERRor:TIME:AUTO:STATe 1035
DISPlay:WLAN:MERRor:TIME:RESet (No Query Form) 1035
DISPlay:WLAN:MERRor:TIME:STARt 1036

DISPlay:WLAN:MERRor:TIME:STOP 1036
DISPlay:WLAN:MERRor:WINDow:SELect:PLOT 1037
DISPlay:WLAN:MERRor:WINDow:TRACe:GRATicule:GRID:STATe 1037
DISPlay:WLAN:MERRor:Y:AUTO (No Query Form) 1038
DISPlay:WLAN:MERRor:Y:OFFSet 1038
DISPlay:WLAN:MERRor:Y:RESet (No Query Form) 1039
DISPlay:WLAN:MERRor:Y[:SCALe] 1039
DISPlay:WLAN:PERRor:FREQuency:AUTO (No Query Form) 1040
DISPlay:WLAN:PERRor:FREQuency:RESet (No Query Form) 1040
DISPlay:WLAN:PERRor:FREQuency:STARt 1040
DISPlay:WLAN:PERRor:FREQuency:STOP 1041
DISPlay:WLAN:PERRor:MARKer:SHOW:STATe 1041
DISPlay:WLAN:PERRor[:SCALe]:AUTO (No Query Form) 1042
DISPlay:WLAN:PERRor:TIME:AUTO (No Query Form) 1042
DISPlay:WLAN:PERRor:TIME:AUTO:STATe 1043
DISPlay:WLAN:PERRor:TIME:RESet (No Query Form) 1043
DISPlay:WLAN:PERRor:TIME:STARt 1044
DISPlay:WLAN:PERRor:TIME:STOP 1044
DISPlay:WLAN:PERRor:WINDow:SELect:PLOT 1045
DISPlay:WLAN:PERRor:WINDow:TRACe:GRATicule:GRID:STATe 1045
DISPlay:WLAN:PERRor:Y:AUTO (No Query Form) 1046
DISPlay:WLAN:PERRor:Y:OFFSet 1046
DISPlay:WLAN:PERRor:Y:RESet (No Query Form) 1047
DISPlay:WLAN:PERRor:Y[:SCALe] 1047
DISPlay:WLAN:PVTime:BURSt:X[:SCALe] 1047
DISPlay:WLAN:PVTime:BURSt:X[:SCALe]:AUTO (No Query Form) 1048
DISPlay:WLAN:PVTime:BURSt:X[:SCALe]:OFFSet 1048
DISPlay:WLAN:PVTime:BURSt:X[:SCALe]:RESet (No Query Form) 1049
DISPlay:WLAN:PVTime:FALL:X[:SCALe] 1049
DISPlay:WLAN:PVTime:FALL:X[:SCALe]:AUTO (No Query Form) 1050
DISPlay:WLAN:PVTime:FALL:X[:SCALe]:OFFSet 1050
DISPlay:WLAN:PVTime:FALL:X[:SCALe]:RESet (No Query Form) 1051
DISPlay:WLAN:PVTime:MARKer:SHOW:STATe 1051
DISPlay:WLAN:PVTime:RISE:X[:SCALe] 1051
DISPlay:WLAN:PVTime:RISE:X[:SCALe]:AUTO (No Query Form) 1052
DISPlay:WLAN:PVTime:RISE:X[:SCALe]:OFFSet 1052
DISPlay:WLAN:PVTime:RISE:X[:SCALe]:RESet (No Query Form) 1053
DISPlay:WLAN:PVTime:WINDow:SELect:PLOT 1053
DISPlay:WLAN:PVTime:WINDow:TRACe:GRATicule:GRID:STATe 1054
DISPlay:WLAN:PVTime:Y[:SCALe] 1054
DISPlay:WLAN:PVTime:Y[:SCALe]:AUTO (No Query Form) 1055
DISPlay:WLAN:PVTime:Y[:SCALe]:OFFSet 1055
DISPlay:WLAN:PVTime:Y[:SCALe]:PDIVision 1056
DISPlay:WLAN:PVTime:Y[:SCALe]:RESet (No Query Form) 1056
DISPlay:WLAN:SUMMary:EVMUnits 1056
*ESE 1057

*ESR? (Query Only) 1057
Fetch commands 1058
FETCh:ACPower? (Query Only) 1058
FETCh:ACPower:CHANnel:POWer? (Query Only) 1059
FETCh:ACPower:SPECtrum? (Query Only) 1059
FETCh:{AM|FM|PM}? (Query Only) 1060
FETCh:AM:AMINdex? (Query Only) 1060
FETCh:AM:AMNegative? (Query Only) 1061
FETCh:AM:AMPositive? (Query Only) 1061
FETCh:AM:RESult? (Query Only) 1062
FETCh:AUDio:FERRor? (Query Only) 1062
FETCh:AUDio:FREQuency? (Query Only) 1063
FETCh:AUDio:HARMonic:COUNt? (Query Only) 1063
FETCh:AUDio:HARMonic<x>:AMPLitude? (Query Only) 1064
FETCh:AUDio:HARMonic<x>:FREQuency? (Query Only) 1064
FETCh:AUDio:HNOise? (Query Only) 1065
FETCh:AUDio:HNREFerence? (Query Only) 1065
FETCh:AUDio:HPTPeak? (Query Only) 1066
FETCh:AUDio:MODDist? (Query Only) 1067
FETCh:AUDio:NHARmonic:COUNt? (Query Only) 1067
FETCh:AUDio:NHARmonic<x>:AMPLitude? (Query Only) 1068
FETCh:AUDio:NHARmonic<x>:FREQuency? (Query Only) 1068
FETCh:AUDio:PNEGative? (Query Only) 1069
FETCh:AUDio:POWer? (Query Only) 1069
FETCh:AUDio:PPOSitive? (Query Only) 1070
FETCh:AUDio:RMS? (Query Only) 1070
FETCh:AUDio:SINad? (Query Only) 1071
FETCh:AUDio:SNOise? (Query Only) 1071
FETCh:AUDio:SPECtrum:TRACe<x>? (Query Only) 1072
FETCh:AUDio:THDist? (Query Only) 1072
FETCh:AUDio:THDist:DB? (Query Only) 1073
FETCh:AUDio:TNHDist? (Query Only) 1073
FETCh:AUDio:TNHDist:DB? (Query Only) 1074
FETCh:AVTime[:MEASview<y>]:AVERage? (Query Only) 1074
FETCh:AVTime[:MEASview<y>]:{FIRSt|SECond|THIRd|FOURth}? (Query Only) 1075
FETCh:AVTime[:MEASview<y>]:MAXimum? (Query Only) 1075
FETCh:AVTime[:MEASview<y>]:MAXLocation? (Query Only) 1076
FETCh:AVTime[:MEASview<y>]:MINimum? (Query Only) 1077
FETCh:AVTime[:MEASview<y>]:MINLocation? (Query Only) 1077
FETCh:AVTime[:MEASview<y>]:RESult? (Query Only) 1078
FETCh:AVTime[:MEASview<y>]:TRACe:BITMap? (Query Only) 1078
:FETCh:AVTime[:MEASview<y>]:AVERage:{FIRSt|SECond|THIRd|FOURth} 1079
:FETCh:AVTime[:MEASview<y>]:MAXimum:{FIRSt|SECond|THIRd|FOURth} 1080
:FETCh:AVTime[:MEASview<y>]:MAXLocation:{FIRSt|SECond|THIRd|FOURth} 1080
:FETCh:AVTime[:MEASview<y>]:MINimum:{FIRSt|SECond|THIRd|FOURth} 1081
FETCh:AVTime[:MEASview<y>]:MINLocation:{FIRSt|SECond|THIRd|FOURth} 1081

FETCh:AVTime[:MEASview<y>]:OGRAm? (Query only) 1082
FETCh:AVTime[:MEASview<y>]:RESult:{FIRSt|SECond|THIRd|FOURth} 1083
FETCh:BIBEmissions:FTX? (Query Only) 1083
FETCh:BIBEmissions:POWer? (Query Only) 1084
FETCh:BIBEmissions:RESUlts:STATus? (Query Only) 1084
FETCh:BLUEtooth:CONSte:FERRor? (Query Only) 1085
FETCh:BLUEtooth:CONSte:TRACe? (Query Only) 1085
FETCh:BLUEtooth:EDIagram:FDEViation? (Query Only) 1086
FETCh:BLUEtooth:EDIagram:FERRor? (Query Only) 1086
FETCh:BLUEtooth:EDIagram:HORiz:SCALe? (Query Only) 1087
FETCh:BLUEtooth:EDIagram:I? (Query Only) 1087
FETCh:BLUEtooth:EDIagram:Q? (Query Only) 1088
FETCh:BLUEtooth:FDVTime:TRACe? (Query Only) 1088
FETCh:BLUEtooth:FREQuency:ERROr? (Query Only) 1089
FETCh:BLUEtooth:FREQuency:ERROr:TYPE? (Query Only) 1089
FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:F1FZero? (Query Only) 1090
FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFN5? (Query Only) 1090
FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFN5:INTErval? (Query Only) 1091
FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFZERO? (Query Only) 1091
FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFZERO:INTErval? (Query Only) 1092
FETCh:BLUEtooth:FREQuency:OFFSet:MAX? (Query Only) 1092
FETCh:BLUEtooth:FREQuency:OFFSet:MAX:INTErval? (Query Only) 1093
FETCh:BLUEtooth:FREQuency:OFFSet:PREAmble? (Query Only) 1093
FETCh:BLUEtooth:MODUlation:CHARacteristics:AVErage:F? (Query Only) 1094
FETCh:BLUEtooth:MODUlation:CHARacteristics:AVErage:F2F1ratio? (Query Only) 1094
FETCh:BLUEtooth:MODUlation:CHARacteristics:AVErage:FONE? (Query Only) 1095
FETCh:BLUEtooth:MODUlation:CHARacteristics:AVErage:FTWO? (Query Only) 1095
FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:COUNt? (Query Only) 1096
FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:F1FZero? (Query Only) 1096
FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:F1FZero: STATus? (Query Only) 1097
FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFN5? (Query Only) 1097
FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFN5: STATus? (Query Only) 1098
FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFZero? (Query Only) 1098
FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFZero: STATus? (Query Only) 1099
FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded: CODIng:SCHEme (Query Only) 1099
FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded: F0FThree (Query Only) 1100
FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded: F0FThree:STATus (Query Only) 1100
FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded: F1FFour (Query Only) 1101
FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded: F1FFour:STATus (Query Only) 1101
FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded: FNFN3 (Query Only) 1102
FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded: FNFN3:STATus (Query Only) 1102
FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:MAX? (Query Only) 1103
FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:MAX:STATus? (Query Only) 1103
FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:PREAmble? (Query Only) 1104
FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:PREAmble:STATus? (Query Only) 1104

FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage: DELTa:F:MAX:PERCentage:COUNt? (Query Only) 1105
FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage: DELTa:FONE:MAX:PERCentage:COUNt? (Query Only) 1105
FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage: DELTa:FTWO:MAX:PERCentage:COUNt? (Query Only) 1106
FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage: DELTa:FTWO:MAX:PERCentage? (Query Only) 1106
FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage: DELTa:FTWO:MAX:PERCentage:STATus? (Query Only) 1107
FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:F? (Query Only) 1107
FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage: F2F1ratio? (Query Only) 1108
FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage: F2F1ratio:STATus? (Query Only) 1108
FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage: FONE? (Query Only) 1109
FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage: FONE:STATus? (Query Only) 1109
FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage: FTWO? (Query Only) 1110
FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:INDex (Query Only) 1110
FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:AVErage? (Query Only) 1111
FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:AVErage:STATus? (Query Only) 1111
FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:COUNt? (Query Only) 1112
FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:EDR:COUNt? (Query Only) 1112
FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:EDR:DPSK? (Query Only) 1113
FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:EDR:GPSK? (Query Only) 1113
FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:PEAK? (Query Only) 1114
FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:PEAK:STATus? (Query Only) 1114
FETCh:BLUEtooth:SUMMary:PACKet:BDR:CRC? (Query Only) 1114
FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:ARQN? (Query Only) 1115
FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:FLOW? (Query Only) 1115
FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:HEC? (Query Only) 1116
FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:LT:ADDRess? (Query Only) 1116
FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:SEQN? (Query Only) 1117
FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:TYPE? (Query Only) 1117
FETCh:BLUEtooth:SUMMary:PACKet:BDR:PAYLoad:LENgth? (Query Only) 1118
FETCh:BLUEtooth:SUMMary:PACKet:BDR:PREAmble? (Query Only) 1118
FETCh:BLUEtooth:SUMMary:PACKet:BDR:SYNC:WORD? (Query Only) 1119
FETCh:BLUEtooth:SUMMary:PACKet:BDR:TYPE? (Query Only) 1119
FETCh:BLUEtooth:SUMMary:PACKet:BLE:ACCEss:ADDRess? (Query Only) 1120
FETCh:BLUEtooth:SUMMary:PACKet:BLE:CRC? (Query Only) 1120
FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:CHANnel: SELect (Query Only) 1121
FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:LENGth? (Query Only) 1121
FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:LLID (Query Only) 1122
FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:MD (Query Only) 1122
FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:NESN (Query Only) 1123
FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:RX:ADDRess? (Query Only) 1123
FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:SN (Query Only) 1124
FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:TX:ADDRess? (Query Only) 1124
FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:TYPE? (Query Only) 1125
FETCh:BLUEtooth:SUMMary:PACKet:BLE:PREAmble? (Query Only) 1125

FETCh:BLUEtooth:SUMMary:PACKet:BLE:TYPE? (Query Only) 1126
FETCh:CCDF? (Query Only) 1126
FETCh:CCDF:{FIRSt|SECond|THIRd}:X? (Query Only) 1127
FETCh:CCDF:{FIRSt|SECond|THIRd}:XY? (Query Only) 1127
FETCh:CCDF:{FIRSt|SECond|THIRd}[:Y]? (Query Only) 1128
FETCh:CONSte[:MEASview<y>]:FERRor? (Query only) 1129
FETCh:CONSte[:MEASview<y>]:RESults? (Query only) 1129
FETCh:CONSte[:MEASview<y>]:TRACe? (Query only) 1130
FETCh:DDEMod[:MEASview<y>]:STABle? (Query only) 1131
FETCh:DDEMod:SYNCh:WORD:LENGth? (Query only) 1131
FETCh:DDEMod:SYNCh:WORD:POSition? (Query Only) 1132
FETCh:DIQVtime[:MEASview<y>]:FERRor? (Query only) 1132
FETCh:DIQVtime[:MEASview<y>]:I? (Query only) 1133
FETCh:DIQVtime[:MEASview<y>]:Q? (Query only) 1134
FETCh:DPX:DDENsity? (Query Only) 1134
FETCh:DPX:DGRam:LINE:COUNt? (Query Only) 1135
FETCh:DPX:DGRam:TIME[:SCALe]:OFFSet? (Query Only) 1135
FETCh:DPX:RESults:TRACe<x>? (Query Only) 1136
FETCh:DPX:TDM:FREQuency:TXBer:BER? (Query Only) 1136
FETCh:DPX:TDM:FREQuency:TXBer:BITS? (Query Only) 1137
FETCh:DPX:TDM:FREQuency:TXBer:ERRors? (Query Only) 1137
FETCh:DPX:TDM:FREQuency:TXBer:STATe? (Query Only) 1138
FETCh:DPX:TRACe:AVERage? (Query Only) 1138
FETCh:DPX:TRACe:BITMap? (Query Only) 1139
FETCh:DPX:TRACe:MATH? (Query Only) 1139
FETCh:DPX:TRACe:MAXimum? (Query Only) 1140
FETCh:DPX:TRACe:MINimum? (Query Only) 1140
FETCh:EDIagram[:MEASview<y>]:FDEViation? (Query only) 1141
FETCh:EDIagram[:MEASview<y>]:FERRor? (Query only) 1142
FETCh:EDIagram[:MEASview<y>]:I? (Query only) 1142
FETCh:EDIagram[:MEASview<y>]:Q? (Query only) 1143
FETCh:EMC:COUNt? (Query only) 1143
FETCh:EMC:EMISsion:RESults:STATus? (Query Only) 1144
FETCh:EMC:SPOT<x>:AMBIent:AMPLitude? (Query Only) 1144
FETCh:EMC:SPOT<x>:AMBIent:DELTa? (Query Only) 1145
FETCh:EMC:SPOT<x>:AMBIent:PASS? (Query Only) 1145
FETCh:EMC:SPOT<x>:FREQuency? (Query Only) 1146
FETCh:EMC:SPOT<x>:PASS? (Query Only) 1146
FETCh:EMC:SPOT<x>:RANGe? (Query Only) 1147
FETCh:EMC:SPOT<x>:RECAll:AMPLitude? (Query Only) 1147
FETCh:EMC:SPOT<x>:RECAll:DELTa? (Query Only) 1148
FETCh:EMC:SPOT<x>:RECAll:PASS? (Query Only) 1148
FETCh:EMC:SPOT<x>:SCDEtector<y>:AMPLitude? (Query Only) 1149
FETCh:EMC:SPOT<x>:SCDEtector<y>:DELTa? (Query Only) 1149
FETCh:EMC:SPOT<x>:SCDEtector<y>:PASS? (Query Only) 1150
FETCh:EMC:SPOT<x>:SPDEtector<y>:AMPLitude? (Query Only) 1151

FETCh:EMC:SPOT<x>:SPDEtector<y>:DELTa? (Query Only) 1151
FETCh:EMC:SPOT<x>:SPDEtector<y>:PASS? (Query Only) 1152
FETCh:EVM[:MEASview<y>]:FERRor? (Query only) 1152
FETCh:EVM[:MEASview<y>]:PEAK? (Query only) 1153
FETCh:EVM[:MEASview<y>]:PINDex? (Query only) 1153
FETCh:EVM[:MEASview<y>]:RMS? (Query only) 1154
FETCh:EVM[:MEASview<y>]:TRACe? (Query only) 1155
FETCh:FDVTime[:MEASview<y>]:FERRor? (Query only) 1155
FETCh:FDVTime[:MEASview<y>]:TRACe? (Query only) 1156
FETCh:{FM|PM}:FERRor? (Query Only) 1156
FETCh:FM:PHALf? (Query Only) 1157
FETCh:FM:PNEGative? (Query Only) 1157
FETCh:FM:PPOSitive? (Query Only) 1158
FETCh:FM:PTPeak? (Query Only) 1158
FETCh:FM:RESult? (Query Only) 1159
FETCh:FM:RMS? (Query Only) 1159
FETCh:{FSETtling|PSETtling}:FTTime? (Query Only) 1160
FETCh:{FSETtling|PSETtling}:MASK[:PASS]? (Query Only) 1160
FETCh:{FSETtling|PSETtling}:SETTled:FREQuency? (Query Only) 1161
FETCh:{FSETtling|PSETtling}:SETTled[:PASS]? (Query Only) 1161
FETCh:{FSETtling|PSETtling}:SETTled:TIME? (Query Only) 1161
FETCh:{FSETtling|PSETtling}:SLMSd[:PASS]? (Query Only) 1162
FETCh:{FSETtling|PSETtling}:STARt:TIME? (Query Only) 1162
FETCh:{FSETtling|PSETtling}:TIME? (Query Only) 1163
FETCh:{FSETtling|PSETtling}:TRACe<x>:X? (Query Only) 1163
FETCh:{FSETtling|PSETtling}:TRACe<x>:XY? (Query Only) 1164
FETCh:{FSETtling|PSETtling}:TRACe<x>[:Y]? (Query Only) 1165
FETCh:{FSETtling|PSETtling}:TRIGger:TIME? (Query Only) 1165
FETCh:{FSETtling|PSETtling}:VALue? (Query Only) 1166
FETCh:FSETtling:ERRor? (Query Only) 1166
FETCh:FVTime[:MEASview<y>]? (Query Only) 1167
FETCh:FVTime[:MEASview<y>]:MAXimum? (Query Only) 1167
FETCh:FVTime[:MEASview<y>]:MAXLocation? (Query Only) 1168
FETCh:FVTime[:MEASview<y>]:MINimum? (Query Only) 1168
FETCh:FVTime[:MEASview<y>]:MINLocation? (Query Only) 1169
FETCh:FVTime[:MEASview<y>]:OGRAm? (Query only) 1169
FETCh:FVTime[:MEASview<y>]:RESult? (Query Only) 1170
FETCh:FVTime[:MEASview<y>]:TRACe:BITMap? (Query Only) 1171
FETCh:IQVTime[:MEASview<y>]:I? (Query Only) 1171
FETCh:IQVTime[:MEASview<y>]:MAXimum (Query Only) 1172
FETCh:IQVTime[:MEASview<y>]:MAXLocation (Query Only) 1172
FETCh:IQVTime[:MEASview<y>]:MINimum (Query Only) 1173
FETCh:IQVTime[:MEASview<y>]:MINLocation (Query Only) 1173
FETCh:IQVTime[:MEASview<y>]:Q (Query Only) 1174
FETCh:IQVTime[:MEASview<y>]:RESult (Query Only) 1174
FETCh:LTE:ACLR:MHITs:COUNt? (Query Only) 1175

FETCh:LTE:ACLR:MHITs<x>:BANDwidth? (Query Only) 1175
FETCh:LTE:ACLR:MHITs<x>:CHANnel:NAME? (Query Only) 1176
FETCh:LTE:ACLR:MHITs<x>:FREQuency:OFFSet? (Query Only) 1176
FETCh:LTE:ACLR:MHITs<x>:INTeg:LOWEr:ABSolute? (Query Only) 1177
FETCh:LTE:ACLR:MHITs<x>:INTeg:LOWEr:RELative? (Query Only) 1177
FETCh:LTE:ACLR:MHITs<x>:INTeg:UPPEr:ABSolute? (Query Only) 1178
FETCh:LTE:ACLR:MHITs<x>:INTeg:UPPEr:RELative? (Query Only) 1178
FETCh:LTE:ACLR:REFerence:POWer? (Query Only) 1178
FETCh:LTE:ACLR:RESUlts:STATus? (Query Only) 1179
FETCh:LTE:ACLR:SPECtrum:X? (Query Only) 1179
FETCh:LTE:ACLR:SPECtrum:XY? (Query Only) 1180
FETCh:LTE:ACLR:SPECtrum:Y? (Query Only) 1180
FETCh:LTE:CHSPectrum:OBW? (Query Only) 1181
FETCh:LTE:CHSPectrum:POWer:CHANnel? (Query Only) 1181
FETCh:LTE:CHSPectrum:SPECtrum? (Query Only) 1182
FETCh:LTE:CONSte:CELL:ID? (Query Only) 1182
FETCh:LTE:CONSte:FREQuency:ERRor? (Query Only) 1182
FETCh:LTE:CONSte:GROUp:ID? (Query Only) 1183
FETCh:LTE:CONSte:SECTor:ID? (Query Only) 1183
FETCh:LTE:PVTime:OFFSlot:POWer? (Query Only) 1184
FETCh:LTE:PVTime:RESUlts:STATus 1184
FETCh:LTE:PVTime:TRACe:X? (Query Only) 1184
FETCh:LTE:PVTime:TRACe:XY? (Query Only) 1185
FETCh:LTE:PVTime:TRACe:Y? (Query Only) 1185
FETCh:MCPower:ADJacent:CHANnels? (Query Only) 1186
FETCh:MCPower:CHANnel:POWer? (Query Only) 1186
FETCh:MCPower:MAIN:CHANnels? (Query Only) 1187
FETCh:MCPower:SPECtrum? (Query Only) 1188
FETCh:MERRor[:MEASview<y>]:FERRor? (Query only) 1188
FETCh:MERRor[:MEASview<y>]:PEAK? (Query only) 1189
FETCh:MERRor[:MEASview<y>]:PINDex? (Query only) 1189
FETCh:MERRor[:MEASview<y>]:RMS? (Query only) 1190
FETCh:MERRor[:MEASview<y>]:TRACe? (Query only) 1190
FETCh:NOISe:GAIN:TRACe<x>:RESult? (Query Only) 1191
FETCh:NOISe:TABLe:VALue? (Query Only) 1191
FETCh:NR:ACP:TAGGpower ? (Query Only) 1192
FETCh:NR:ACP:NO:OF:NR:OFFsets? (Query Only) 1192
FETCh:NR:ACP:ADJChannel<x>:INTBw? (Query Only) 1193
FETCh:NR:ACP:ADJChannel<x>:UABS? (Query Only) 1193
FETCh:NR:ACP:ADJChannel<x>:UREL ? (Query Only) 1193
FETCh:NR:ACP:ADJChannel<x>:LABS? (Query Only) 1194
FETCh:NR:ACP:ADJChannel<x>:LREL ? (Query Only) 1194
FETCh:NR:ACP:ADJChannel<x>:FREQoffset ? (Query Only) 1195
FETCh:NR:CHP:CARRier<x>:ABSPower? (Query Only) 1195
FETCh:NR:CHP:CARRier<x>:RELPower? (Query Only) 1196
FETCh:NR:CHP:TAGGpower? (Query Only) 1196

FETCh:NR:EVM:ATSYmbol (Query Only) 1196
FETCh:NR:EVM:PEAK (Query Only) 1197
FETCh:NR:EVM:RMS (Query Only) 1197
FETCh:NR:LINK:DIREction? (Query Only) 1197
FETCh:NR:MODAcc:CARRier<x>:CELL:ID? (Query Only) 1198
FETCh:NR:MODAcc:CARRier<x>:OFFSet? (Query Only) 1198
FETCh:NR:MODAcc:CARRier<x>:CREM? (Query Only) 1199
FETCh:NR:MODAcc:CARRier<x>:CPEM? (Query Only) 1199
FETCh:NR:MODAcc:CARRier<x>:CPESLI? (Query Only) 1199
FETCh:NR:MODAcc:CARRier<x>:CPESYI? (Query Only) 1200
FETCh:NR:MODAcc:CARRier<x>:CPESCI? (Query Only) 1200
FETCh:NR:MODAcc:CARRier<x>:CCFME? (Query Only) 1201
FETCh:NR:MODAcc:CARRier<x>:CCIOOM? (Query Only) 1201
FETCh:NR:MODAcc:CARRier<x>:CCIGIM? (Query Only) 1202
FETCh:NR:MODAcc:CARRier<x>:CCIQEM? (Query Only) 1202
FETCh:NR:MODAcc:CARRier<x>:IBEM? (Query Only) 1202
FETCh:NR:MODAcc:CARRier<x>:DATA:PEAK (Query Only) 1203
FETCh:NR:MODAcc:CARRier<x>:DATA:RMS (Query Only) 1203
FETCh:NR:MODAcc:CARRier<x>:DMRS:PEAK (Query Only) 1204
FETCh:NR:MODAcc:CARRier<x>:DMRS:RMS (Query Only) 1204
FETCh:NR:MODAcc:CARRier<x>:PREM (Query Only) 1204
FETCh:NR:MODAcc:CARRier<x>:PTRS: RMS (Query Only) 1205
FETCh:NR:MODAcc:CARRier<x>:PTRS:PEAK (Query Only) 1205
FETCh:NR:OBW:ABSPower (Query Only) 1206
FETCh:NR:OBW:OCCUpiedbw (Query Only) 1206
FETCh:NR:OBW:STARtfreq (Query Only) 1207
FETCh:NR:OBW:STOPfreq (Query Only) 1207
FETCh:NR:PVT:ABS:OFFpower:AFTR (Query Only) 1207
FETCh:NR:PVT:ABS:OFFpower:BFR (Query Only) 1208
FETCh:NR:PVT:ABS:ONpower (Query Only) 1208
FETCh:NR:PVT:BURStwidth (Query Only) 1208
FETCh:NR:PVT:MEASurement:STATus (Query Only) 1209
FETCh:NR:SEM:TAGGpower? (Query Only) 1209
FETCh:NR:SEM:OFFSet<x>:STARtfreq? (Query Only) 1210
FETCh:NR:SEM:OFFSet<x>:STOPfreq? (Query Only) 1210
FETCh:NR:SEM:OFFSet<x>:PEAK:ABS? (Query Only) 1210
FETCh:NR:SEM:OFFSet<x>:PEAK:REL? (Query Only) 1211
FETCh:NR:SEM:OFFSet<x>:PEAK:FREQ? (Query Only) 1211
FETCh:NR:SEM:OFFSet<x>:MARGin? (Query Only) 1212
FETCh:NR:SEM:OFFSet<x>:MARGin:ABS? (Query Only) 1212
FETCh:NR:SEM:OFFSet<x>:MARGin:REL? (Query Only) 1212
FETCh:NR:SEM:OFFSet<x>:MARGin:FREQ? (Query Only) 1213
FETCh:NR:SEM:OFFSet<x>:INTEg:ABS? (Query Only) 1213
FETCh:NR:SEM:OFFSet<x>:INTEg:REL? (Query Only) 1214
FETCh:NR:SEM:OFFSet<x>:RBW? (Query Only) 1214
FETCh:NR:SEM:OFFSet<x>:ABSLimit:STARt? (Query Only) 1214

FETCh:NR:SEM:OFFSet<x>:ABSLimit:STOP? (Query Only) 1215
FETCh:NR:SEM:OFFSet<x>:RELLimit:STARt? (Query Only) 1215
FETCh:NR:SEM:OFFSet<x>:RELLimit:STOP? (Query Only) 1216
FETCh:NR:SUBblock<x>:NUMCcs? (Query Only) 1216
FETCh:OBWidth:BOBW:FREQuency:ERRor (Query Only) 1216
FETCh:OBWidth:BOBW:RESUlts:STATus (Query Only) 1217
FETCh:OBWidth:BOBW:XDBB:POWer (Query Only) 1217
FETCh:OBWidth:BOBW:XDBBandwidth? (Query Only) 1218
FETCh:OBWidth:BOBW:XDBMbandwidth:IN:BANDwidth? (Query Only) 1218
FETCh:OBWidth:BOBW:XDBMbandwidth:IN:LEFT:FREQuency? (Query Only) 1219
FETCh:OBWidth:BOBW:XDBMbandwidth:IN:RIGHt:FREQuency? (Query Only) 1219
FETCh:OBWidth:BOBW:XDBMbandwidth:OUT:BANDwidth? (Query Only) 1220
FETCh:OBWidth:BOBW:XDBMbandwidth:OUT:LEFT:FREQuency? (Query Only) 1221
FETCh:OBWidth:BOBW:XDBMbandwidth:OUT:RIGHt:FREQuency? (Query Only) 1221
FETCh:OBWidth:FREQuency:ERRor? (Query Only) 1222
FETCh:OBWidth:OBWidth:BANDwidth? (Query Only) 1222
FETCh:OBWidth:OBWidth:LEFT:FREQuency? (Query Only) 1223
FETCh:OBWidth:OBWidth:LEFT:LEVel? (Query Only) 1223
FETCh:OBWidth:OBWidth:POWer? (Query Only) 1224
FETCh:OBWidth:OBWidth:RIGHt:FREQuency? (Query Only) 1224
FETCh:OBWidth:OBWidth:RIGHt:LEVel? (Query Only) 1225
FETCh:OBWidth:SPECtrum? (Query Only) 1225
FETCh:OBWidth:XDBBandwidth:BANDwidth? (Query Only) 1226
FETCh:OBWidth:XDBBandwidth:LEFT:FREQuency? (Query Only) 1226
FETCh:OBWidth:XDBBandwidth:LEFT:LEVel? (Query Only) 1227
FETCh:OBWidth:XDBBandwidth:POWer? (Query Only) 1227
FETCh:OBWidth:XDBBandwidth:RIGHt:FREQuency? (Query Only) 1228
FETCh:OBWidth:XDBBandwidth:RIGHt:LEVel? (Query Only) 1228
FETCh:OFDM:APOWer? (Query Only) 1229
FETCh:OFDM:APOWer:PEAK? (Query Only) 1229
FETCh:OFDM:CONSte:MAGNitude? (Query Only) 1230
FETCh:OFDM:CONSte:PHASe? (Query Only) 1230
FETCh:OFDM:CONSte:TYPE? (Query Only) 1231
FETCh:OFDM:CONSte:VALue? (Query Only) 1232
FETCh:OFDM:CPE? (Query Only) 1232
FETCh:OFDM:CRESponse:MAGNitude? (Query Only) 1233
FETCh:OFDM:CRESponse:PHASe? (Query Only) 1233
FETCh:OFDM:EVM:PEAK:DECibel:ALL? (Query Only) 1234
FETCh:OFDM:EVM:PEAK:DECibel:DATA? (Query Only) 1234
FETCh:OFDM:EVM:PEAK:DECibel:PILots? (Query Only) 1235
FETCh:OFDM:EVM:PEAK:PERCent:ALL? (Query Only) 1235
FETCh:OFDM:EVM:PEAK:PERCent:DATA? (Query Only) 1236
FETCh:OFDM:EVM:PEAK:PERCent:PILots? (Query Only) 1236
FETCh:OFDM:EVM:PEAK:SCARrier:ALL? (Query Only) 1237
FETCh:OFDM:EVM:PEAK:SCARrier:DATA? (Query Only) 1237
FETCh:OFDM:EVM:PEAK:SCARrier:PILots? (Query Only) 1238

FETCh:OFDM:EVM:PEAK:SYMBol:ALL? (Query Only) 1238
FETCh:OFDM:EVM:PEAK:SYMBol:DATA? (Query Only) 1238
FETCh:OFDM:EVM:PEAK:SYMBol:PILots? (Query Only) 1239
FETCh:OFDM:EVM:RMS:DECibel:ALL? (Query Only) 1239
FETCh:OFDM:EVM:RMS:DECibel:DATA? (Query Only) 1240
FETCh:OFDM:EVM:RMS:DECibel:PILots? (Query Only) 1240
FETCh:OFDM:EVM:RMS:PERCent:ALL? (Query Only) 1241
FETCh:OFDM:EVM:RMS:PERCent:DATA? (Query Only) 1241
FETCh:OFDM:EVM:RMS:PERCent:PILots? (Query Only) 1242
FETCh:OFDM:EVM:TRACe<x>? (Query Only) 1242
FETCh:OFDM:FERRor? (Query Only) 1243
FETCh:OFDM:FLATness:PASS? (Query Only) 1243
FETCh:OFDM:FLATness:RESult? (Query Only) 1244
FETCh:OFDM:FLATness:TRACe<x>? (Query Only) 1244
FETCh:OFDM:GAIN:IMBalance? (Query Only) 1245
FETCh:OFDM:IQ:ORIGin:OFFSet? (Query Only) 1245
FETCh:OFDM:MERRor:TRACe<x>? (Query Only) 1246
FETCh:OFDM:PACKet:DIRection? (Query Only) 1246
FETCh:OFDM:PERRor:TRACe<x>? (Query Only) 1247
FETCh:OFDM:POWer:TRACe<x>? (Query Only) 1248
FETCh:OFDM:QUADrature:OFFSet? (Query Only) 1248
FETCh:OFDM:SCARriers? (Query Only) 1249
FETCh:OFDM:SCARriers:SPACing? (Query Only) 1249
FETCh:OFDM:STABle:VALUe? (Query Only) 1250
FETCh:OFDM:SYMBol:CERRor? (Query Only) 1250
FETCh:OFDM:SYMBol:COUNt? (Query Only) 1251
FETCh:P25:CONSte:FERRor? (Query Only) 1251
FETCh:P25:CONSte:TRACe? (Query Only) 1251
FETCh:P25:EDIagram:FDEViation? (Query Only) 1252
FETCh:P25:EDIagram:FERRor? (Query Only) 1252
FETCh:P25:EDIagram:I? (Query Only) 1253
FETCh:P25:EDIagram:Q? (Query Only) 1253
FETCh:P25:PVTime:BURSt:WIDTh? (Query Only) 1254
FETCh:P25:PVTime:FALL:TIME:STARt? (Query Only) 1254
FETCh:P25:PVTime:FALL:TIME:STOP? (Query Only) 1255
FETCh:P25:PVTime:RISE:TIME? (Query Only) 1255
FETCh:P25:PVTime:RISE:TIME:STARt? (Query Only) 1256
FETCh:P25:PVTime:RISE:TIME:STOP? (Query Only) 1256
FETCh:P25:PVTime:RTIMe? (Query Only) 1257
FETCh:P25:PVTime:TRACe[:Y]? (Query Only) 1257
FETCh:P25:SUMMary:LIMIt: SRA (Query Only) 1258
FETCh:P25:SUMMary:MODUlation:MEASurement:FIDelity (Query Only) 1258
FETCh:P25:SUMMary:MODUlation:MEASurement:FIDelity:STATus (Query Only) 1259
FETCH:P25:SUMMary:MODUlation:MEASurement: FREQuency:.DEVIation:POSPeak 1259
FETCh:P25:SUMMary:MODUlation:MEASurement: FREQuency:.DEVIation:POSPeak:STA 1260
FETCH:P25:SUMMary:MODUlation:MEASurement:FREQuency: ACCUracy (Query Only) 1260

FETCH:P25:SUMMary:MODUlation:MEASurement:FREQuency: ACCUracy (Query Only) 1261
FETCh:P25:SUMMary:MODUlation:MEASurement:FREQuency: ACCUracy:STATus (Query Only) 1261
FETCh:P25:SUMMary:MODUlation:MEASurement:FREQuency: DEVIation:NEGPeak:STATus (Query Only) 1262
FETCH:P25:SUMMary:MODUlation:MEASurement:FREQuency: DEVIation:NEGPeak (Query Only) 1262
FETCh:P25:SUMMary:MODUlation:MEASurement:FREQuency: DEVIation:POSPeak:STATus (Query Only) 1263
FETCH:P25:SUMMary:MODUlation:MEASurement:FREQuency: DEVIation:POSPeak (Query Only) 1263
FETCH:P25:SUMMary:MODUlation:MEASurement:FREQuency:ERRor (Query Only) 1264
FETCH:P25:SUMMary:MODUlation:MEASurement:SRA? (Query Only) 1264
FETCh:P25:SUMMary:MODUlation:MEASurement:SRA:STATus (Query Only) 1265
FETCH:P25:SUMMary:POWEr:MEASurement:HCPM:MAXOFF? (Query Only) 1265
FETCh:P25:SUMMary:POWEr:MEASurement:HCPM:MAXOFF:STATus? (Query 1265
FETCH:P25:SUMMary:POWEr:MEASurement:HCPM:MAXON? (Query Only) 1266
FETCh:P25:SUMMary:POWEr:MEASurement:HCPM:MAXON:STATus? (Query Only) 1266
FETCH:P25:SUMMary:POWEr:MEASurement:HCPM:MAXSS? (Query Only) 1267
FETCh:P25:SUMMary:POWEr:MEASurement:HCPM:MAXSs:STATus (Query Only) 1267
FETCH:P25:SUMMary:POWEr:MEASurement:HCPM:MINImum? (Query Only) 1268
FETCh:P25:SUMMary:POWEr:MEASurement:HCPM:MINImum:STATus? (Query Only) 1268
FETCH:P25:SUMMary:POWEr:MEASurement:HCPM:OFFSlot? (Query Only) 1268
FETCh:P25:SUMMary:POWEr:MEASurement:HCPM:OFFSlot:STATus? (Query Only) 1269
FETCh:P25:SUMMary:POWEr:MEASurement:HCPM:PEAK:ACPR:HI (Query Only) 1269
FETCh:P25:SUMMary:POWEr:MEASurement:HCPM:PEAK:ACPR:LOW (Query 1270
FETCh:P25:SUMMary:POWEr:MEASurement:HCPM:PEAK:ACPR:MIN (Query 1270
FETCh:P25:SUMMary:POWEr:MEASurement:HCPM:PEAK:ACPR: MIN:STATus 1271
FETCh:P25:SUMMary:POWEr:MEASurement:RF (Query Only) 1271
FETCh:P25:SUMMary:POWEr:MEASurement:RF:STATus (Query Only) 1271
FETCh:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment: TERRor:ONE:.ACQ:COUNT (Query Only) 1272
FETCh:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment: TERRor:ONE:ACQ:COUNT (Query Only) 1272
FETCh:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment: TERRor:ZERO:STATus (Query Only) 1273
FETCh:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment: TERRor:ONE:STATus (Query Only) 1273
FETCh:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment: TERRor:ZERO:ACQ:COUNT (Query Only) 1274
FETCh:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME: ALIGNment:TERRor:.ZERO:S(Query Only) 1274
FETCH:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment: TERRor:ONE:.ACQ:COUNT?(Query Only) 1275
FETCH:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment: TERRor:ONE:?(Query Only) 1275
FETCH:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment: TERRor:ZERO? 1276
FETCH:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment: TOBSync? (Query Only) 1276
FETCh:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment: TOBSync:ACQ:COUNT (Query Only) 1277
FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck: TIME:.ENCOder? (Query Only) 1277
FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck: TIME:.ENCOder:BI:STAT(Query Only) 1277
FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck: TIME:ENCOder: STATus (Query Only) 1278
FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck: TIME:POWEr:BI (Query Only) 1279
FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck: TIME:ACQ:COUNT(Query Only) 1279
FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck: TIME:ACQ:COUNT:BI(Query Only) 1279

FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck: TIME:ENCOder:BI:STATus (Query Only) 1280
FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck: TIME:ENCOder: STATus (Query Only) 1281
FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck: TIME:ENCOder?(Query Only) 1281
FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck: TIME:ENCOder:BI?(Query Only) 1282
FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck: TIME:POWEr (Query Only) 1282
FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck: TIME:POWEr:BI (Query Only) 1282
FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck: TIME:POWEr:BI:STATus(Query Only) 1283
FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck: TIME:POWEr:BI?(Query Only) 1283
FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck: TIME: POWEr:STATus (Query Only) 1284
FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck: TIME:POWEr?(Query Only) 1284
FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput: DELAy (Query Only) 1285
FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput: DELAy:ACQ:COUNT (Query Only) 1285
FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput: DELAy:ACQ:COUNT?(Query Only) 1285
FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput: DELAy:STATus (Query Only) 1286
FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput: DELAy? (Query Only) 1286
FETCh:PERRor[:MEASview<y>]:FERRor? (Query only) 1287
FETCh:PERRor[:MEASview<y>]:PEAK? (Query only) 1287
FETCh:PERRor[:MEASview<y>]:PINDex? (Query only) 1288
FETCh:PERRor[:MEASview<y>]:RMS? (Query only) 1288
FETCh:PERRor[:MEASview<y>]:TRACe? (Query only) 1289
FETCh:PHVTime[:MEASview<y>]? (Query Only) 1290
FETCh:PHVTime[:MEASview<y>]:MAXimum? (Query Only) 1290
FETCh:PHVTime[:MEASview<y>]:MAXLocation? (Query Only) 1291
FETCh:PHVTime[:MEASview<y>]:MINimum? (Query Only) 1291
FETCh:PHVTime[:MEASview<y>]:MINLocation? (Query Only) 1292
FETCh:PHVTime[:MEASview<y>]:OGRAm? (Query only) 1292
FETCh:PHVTime[:MEASview<y>]:RESult? (Query Only) 1293
FETCh:PHVTime[:MEASview<y>]:TRACe:BITMap? (Query Only) 1294
FETCh:PM:PNEGative? (Query Only) 1294
FETCh:PM:PPOSitive? (Query Only) 1295
FETCh:PM:PTPeak? (Query Only) 1295
FETCh:PM:RESult? (Query Only) 1295
FETCh:PM:RMS? (Query Only) 1296
FETCh:PNOise:ALL? (Query Only) 1296
FETCh:PNOise:CARRier:FERRor? (Query Only) 1297
FETCh:PNOise:CARRier:POWer? (Query Only) 1298
FETCh:PNOise:JITTer? (Query Only) 1298
FETCh:PNOise:RESidual:FM? (Query Only) 1298
FETCh:PNOise:RMS:PNOise? (Query Only) 1299
FETCh:PNOise:SPECtrum<x>:X? (Query Only) 1299
FETCh:PNOise:SPECtrum<x>:XY? (Query Only) 1300
FETCh:PNOise:SPECtrum<x>[:Y]? (Query Only) 1301
FETCh:PULSe:CUMulative:HISTogram[:MEASview<y>]: OUTLier:HIGHer? (Query Only) 1301
FETCh:PULSe:CUMulative:HISTogram[:MEASview<y>]: OUTLier:LOWer? (Query Only) 1302
FETCh:PULSe:CUMulative:HISTogram[:MEASview<y>]:PULse:COUNt? (Query Only) 1302
FETCh:PULSe:CUMulative:HISTogram[:MEASview<y>]:PULse:TOTal? (Query Only) 1303

FETCh:PULSe:CUMulative:STATistics[:MEASview<y>] 1303
FETCh:PULSe:CUMulative:STATistics[:MEASview<y>] (Query Only) 1305
FETCh:PULSe:OGRAm[:MEASview<y>]:MAX:FRAMe:COUNt (Query Only) 1306
FETCh:PULSe:OGRAm[:MEASview<y>]:RBW (Query Only) 1307
FETCh:PULSe[:RESult][:MEASview<y>]:ADVanced (Query Only) 1307
FETCh:PULSe[:RESult][:MEASview<y>]:ADVanced:RANGe (Query only) 1309
FETCh:PULSe[:RESult][:MEASview<y>]:ATX? (Query Only) 1311
FETCh:PULSe[:RESult][:MEASview<y>]:AVERage? (Query Only) 1311
FETCh:PULSe[:RESult][:MEASview<y>]:DRODb? (Query Only) 1312
FETCh:PULSe[:RESult][:MEASview<y>]:DROop? (Query Only) 1312
FETCh:PULSe[:RESult][:MEASview<y>]:DUTPct? (Query Only) 1313
FETCh:PULSe[:RESult][:MEASview<y>]:DUTRatio? (Query Only) 1314
FETCh:PULSe[:RESult][:MEASview<y>]:FABS? (Query Only) 1314
FETCh:PULSe[:RESult][:MEASview<y>]:FALL? (Query Only) 1315
FETCh:PULSe[:RESult][:MEASview<y>]:FDELta? (Query Only) 1315
FETCh:PULSe[:RESult][:MEASview<y>]:FRDeviation? (Query Only) 1316
FETCh:PULSe[:RESult][:MEASview<y>]:IRAMplitude? (Query Only) 1316
FETCh:PULSe[:RESult][:MEASview<y>]:IRTime? (Query Only) 1317
FETCh:PULSe[:RESult][:MEASview<y>]:MFREqerror? (Query Only) 1318
FETCh:PULSe[:RESult][:MEASview<y>]:MPHerror? (Query Only) 1318
FETCh:PULSe[:RESult][:MEASview<y>]:OVEDb? (Query Only) 1319
FETCh:PULSe[:RESult][:MEASview<y>]:OVERshoot? (Query Only) 1319
FETCh:PULSe[:RESult][:MEASview<y>]:PHDeviation? (Query Only) 1320
FETCh:PULSe[:RESult][:MEASview<y>]:PPFD? (Query Only) 1320
FETCh:PULSe[:RESult][:MEASview<y>]:PPFRequency? (Query Only) 1321
FETCh:PULSe[:RESult][:MEASview<y>]:PPOWer? (Query Only) 1322
FETCh:PULSe[:RESult][:MEASview<y>]:PPPD? (Query Only) 1322
FETCh:PULSe[:RESult][:MEASview<y>]:PPPHase? (Query Only) 1323
FETCh:PULSe[:RESult]:MEASview<y>:RINTerval? (Query Only) 1323
FETCh:PULSe[:RESult][:MEASview<y>]:RIPDb? (Query Only) 1324
FETCh:PULSe[:RESult][:MEASview<y>]:RIPPle? (Query Only) 1325
FETCh:PULSe[:RESult][:MEASview<y>]:RISE? (Query Only) 1325
FETCh:PULSe[:RESult][:MEASview<y>]:RMSFreqerror? (Query Only) 1326
FETCh:PULSe[:RESult][:MEASview<y>]:RMSPherror? (Query Only) 1326
FETCh:PULSe[:RESult][:MEASview<y>]:RRATe? (Query Only) 1327
FETCh:PULSe[:RESult][:MEASview<y>]:TIME? (Query Only) 1327
FETCh:PULSe[:RESult][:MEASview<y>]:WIDTh? (Query Only) 1328
FETCH:PULSe:STATistics[:MEASview<y>]:ADVanced (Query Only) 1329
FETCh:PULSe:STATistics[:MEASview<y>]:ATX (Query Only) 1330
FETCh:PULSe:STATistics[:MEASview<y>]:AVERage (Query Only) 1331
FETCh:PULSe:STATistics[:MEASview<y>]:DRODb (Query Only) 1331
FETCh:PULSe:STATistics[:MEASview<y>]:DROop (Query Only) 1332
FETCh:PULSe:STATistics[:MEASview<y>]:DUTPct (Query Only) 1333
FETCh:PULSe:STATistics[:MEASview<y>]:DUTRatio (Query Only) 1333
FETCh:PULSe:STATistics[:MEASview<y>]:FABS (Query Only) 1334
FETCh:PULSe:STATistics[:MEASview<y>]:FALL (Query Only) 1334

FETCh:PULSe:STATistics[:MEASview<y>]:FDELta (Query Only) 1335
FETCh:PULSe:STATistics[:MEASview<y>]:FRDeviation (Query Only) 1336
FETCh:PULSe:STATistics[:MEASview<y>]:IRAMplitude (Query Only) 1336
FETCh:PULSe:STATistics[:MEASview<y>]:IRTime (Query Only) 1337
FETCH:PULSe:STATistics[:MEASview<y>]:MAX:PULSe:NUMber (Query Only) 1337
FETCh:PULSe:STATistics[:MEASview<y>]:MFReqerror (Query Only) 1338
FETCH:PULSe:STATistics[:MEASview<y>]:MIN:PULSe:NUMber (Query Only) 1338
FETCh:PULSe:STATistics[:MEASview<y>]:MPHerror (Query Only) 1339
FETCh:PULSe:STATistics[:MEASview<y>]:OVEDb (Query Only) 1340
FETCh:PULSe:STATistics[:MEASview<y>]:OVERshoot (Query Only) 1340
FETCh:PULSe:STATistics[:MEASview<y>]:PHDeviation (Query Only) 1341
FETCh:PULSe:STATistics[:MEASview<y>]:PPFD (Query Only) 1342
FETCh:PULSe:STATistics[:MEASview<y>]:PPFRequency (Query Only) 1342
FETCh:PULSe:STATistics[:MEASview<y>]:PPOWer (Query Only) 1343
FETCh:PULSe:STATistics[:MEASview<y>]:PPPD (Query Only) 1343
FETCh:PULSe:STATistics[:MEASview<y>]:PPPHase (Query Only) 1344
FETCh:PULSe:STATistics[:MEASview<y>]:RINTerval (Query Only) 1344
FETCh:PULSe:STATistics[:MEASview<y>]:RIPDb (Query Only) 1345
FETCh:PULSe:STATistics[:MEASview<y>]:RIPPle (Query Only) 1346
FETCh:PULSe:STATistics[:MEASview<y>]:RISE (Query Only) 1346
FETCh:PULSe:STATistics[:MEASview<y>]:RMSFreqerror (Query Only) 1347
FETCh:PULSe:STATistics[:MEASview<y>]:RMSPherror (Query Only) 1348
FETCh:PULSe:STATistics[:MEASview<y>]:RRATe (Query Only) 1348
FETCh:PULSe:STATistics[:MEASview<y>]:WIDTh (Query Only) 1349
FETCh:PULSe:STATistics[:MEASview<y>]:X? (Query Only) 1349
FETCh:PULSe:STATistics[:MEASview<y>]:XY? (Query Only) 1350
FETCh:PULSe:STATistics[:MEASview<y>][:Y]? (Query Only) 1351
FETCh:PULSe:TRACe[:MEASview<y>]:X 1352
FETCh:PULSe:TRACe[:MEASview<y>]:XY 1352
FETCh:PULSe:TRACe[:MEASview<y>]:Y 1353
FETCh:RFIN:IQ? (Query Only) 1354
FETCh:RFIN:IQ:HEADer? (Query Only) 1355
FETCh:RFIN:IQ:SCALe? (Query Only) 1355
FETCh:RFIN:IQ:VHEader? (Query Only) 1356
FETCh:RFIN:RECord:IDS? (Query Only) 1357
FETCh:SEM:MHITs:COUNt? (Query Only) 1357
FETCh:SEM:MHITs<x>:AMPLitude:ABSolute? (Query Only) 1358
FETCh:SEM:MHITs<x>:AMPLitude:RELative? (Query Only) 1358
FETCh:SEM:MHITs<x>:FREQuency? (Query Only) 1359
FETCh:SEM:MHITs<x>:INTeg:ABSolute? (Query Only) 1359
FETCh:SEM:MHITs<x>:INTeg:RELative? (Query Only) 1360
FETCh:SEM:MHITs<x>:MARGin:ABSolute? (Query Only) 1360
FETCh:SEM:MHITs<x>:MARGin:RELative? (Query Only) 1361
FETCh:SEM:MHITs<x>:OFFSet? (Query Only) 1361
FETCh:SEM:PASS? (Query Only) 1362
FETCh:SEM:REF:POWer? (Query Only) 1362

FETCh:SEM:SPECtrum:X? (Query Only) 1363
FETCh:SEM:SPECtrum:XY? (Query Only) 1363
FETCh:SEM:SPECtrum[:Y]? (Query Only) 1364
FETCh:SGRam[:MEASview<y>] (Query Only) 1364
FETCh:SIGNalvu:ACQuisition:COUPling? (Query Only) 1365
FETCh:SIGNalvu:ACQuisition:MAX:RT:BW? (Query Only) 1365
FETCH:SIGNalvu:ACQuisition[:SOURce<x>]: DOWNconverter:INJEction:HIGHSide:MAXFreq? (Query only) 1366
FETCH:SIGNalvu:ACQuisition[:SOURce<x>]: DOWNconverter:INJEction:HIGHSide:MINFreq? (Query only) 1366
FETCH:SIGNalvu:ACQuisition[:SOURce<x>]: DOWNconverter:INJEction:LOWSide:MAXFreq? (Query only) 1367
FETCH:SIGNalvu:ACQuisition[:SOURce<x>]: DOWNconverter:INJEction:LOWSide:MINFreq? (Query only) 1368
FETCh:SIGNalvu:ACQuisition:TERMination? (Query Only) 1368
FETCh:SPECtrum[:MEASview<y>]:TRACe:BITMap? (Query Only) 1369
FETCh:SPECtrum[:MEASview<y>]:TRACe<x>? (Query Only) 1369
FETCh:SPURious:CARRier:POWer? (Query Only) 1370
FETCh:SPURious:COUNt? (Query Only) 1370
FETCh:SPURious:PASS? (Query Only) 1371
FETCh:SPURious:SPECtrum:X? (Query Only) 1371
FETCh:SPURious:SPECtrum:XY? (Query Only) 1372
FETCh:SPURious:SPECtrum[:Y]? (Query Only) 1372
FETCh:SPURious:SPUR<x>:AMPLitude:ABSolute? (Query Only) 1373
FETCh:SPURious:SPUR<x>:AMPLitude:RELative? (Query Only) 1373
FETCh:SPURious:SPUR<x>:FREQuency:ABSolute? (Query Only) 1374
FETCh:SPURious:SPUR<x>:FREQuency:RELative? (Query Only) 1374
FETCh:SPURious:SPUR<x>:LIMit:ABSolute? (Query Only) 1375
FETCh:SPURious:SPUR<x>:LIMit:RELative? (Query Only) 1375
FETCh:SPURious:SPUR<x>:LIMit:VIOLation? (Query Only) 1376
FETCh:SPURious:SPUR<x>:RANGe? (Query Only) 1376
FETCh:SQUality[:MEASview<y>]:FREQuency:DEViation? (Query only) 1377
FETCh:SQUality[:MEASview<y>]:FREQuency:DEViation:TABLe? (Query only) 1377
FETCh:SQUality[:MEASview<y>]:FREQuency:ERRor? (Query only) 1378
FETCh:SQUality[:MEASview<y>]:GAIN:IMBalance? (Query only) 1379
FETCh:SQUality[:MEASview<y>]:ORIGin:OFFSet? (Query only) 1379
FETCh:SQUality[:MEASview<y>]:PEAK:EVM? (Query only) 1380
FETCh:SQUality[:MEASview<y>]:PEAK:EVM:DB? (Query only) 1380
FETCh:SQUality[:MEASview<y>]:PEAK:EVM:DB:OFFSet? (Query only) 1381
FETCh:SQUality[:MEASview<y>]:PEAK:EVM:LOCation? (Query only) 1382
FETCh:SQUality[:MEASview<y>]:PEAK:EVM:LOCation:OFFSet? (Query only) 1382
FETCh:SQUality[:MEASview<y>]:PEAK:EVM:OFFSet? (Query only) 1383
FETCh:SQUality [:MEASview<y>]:PEAK:FERRor? (Query only) 1383
FETCh:SQUality[:MEASview<y>] :PEAK:MERRor:DB? (Query only) 1384
FETCh:SQUality[:MEASview<y>]:PEAK:MERRor:LOCation? (Query only) 1385
FETCh:SQUality[:MEASview<y>]:PEAK:MERRor? (Query only) 1385
FETCh:SQUality[:MEASview<y>]:PEAK:PERRor:LOCation? (Query only) 1386
FETCh:SQUality[:MEASview<y>]:PEAK:PERRor? (Query only) 1386
FETCh:SQUality[:MEASview<y>]:QUADrature:ERRor? (Query only) 1387
FETCh:SQUality [:MEASview<y>]:RHO? (Query only) 1387

FETCh:SQUality[:MEASview<y>]:RMS:EVM? (Query only) 1388
FETCh:SQUality[:MEASview<y>]:RMS:EVM:DB? (Query only) 1388
FETCh:SQUality[:MEASview<y>]:RMS:EVM:DB:OFFSet? (Query only) 1389
FETCh:SQUality[:MEASview<y>]:RMS:EVM:OFFSet? (Query only) 1390
FETCh:SQUality[:MEASview<y>]:RMS:FERRor? (Query only) 1390
FETCh:SQUality[:MEASview<y>]:RMS:MER:DB? (Query only) 1391
FETCh:SQUality[:MEASview<y>]:RMS:MERRor? (Query only) 1391
FETCh:SQUality[:MEASview<y>]:RMS:MERRor:DB? (Query only) 1392
FETCh:SQUality[:MEASview<y>]:RMS:PERRor? (Query only) 1392
FETCh:SQUality[:MEASview<y>]:SYMBol:LENGth? (Query only) 1393
FETCh:SQUality[:MEASview<y>]:SYMBol:RATE? (Query only) 1393
FETCh:SQUality[:MEASview<y>]:SYMBol:RATE:ERRor? (Query only) 1394
FETCh:TDIagram[:MEASview<y>] :FERRor? (Query only) 1395
FETCh:TDIagram[:MEASview<y>]:TRACe? (Query only) 1395
FETCh:TOVerview[:MEASview<y>] (Query Only) 1396
FETCh:TXGain:AVERage? (Query Only) 1396
FETCh:TXGain:MAX? (Query Only) 1397
FETCh:TXGain:MAXLocation? (Query Only) 1397
FETCh:TXGain:MIN? (Query Only) 1397
FETCh:TXGain:MINLocation? (Query Only) 1398
FETCh:TXGain:TRACe<x>? (Query Only) 1398
FETCh:WLAN:AD:CONSte:DATA:EVM? (Query Only) 1399
FETCh:WLAN:AD:CONSte:DATA:EVM:PEAK? (Query Only) 1399
FETCh:WLAN:AD:CONSte:DATA:EVM:PEAK:SYMBol? (Query Only) 1400
FETCh:WLAN:AD:CONSte:FERRor? (Query Only) 1400
FETCh:WLAN:AD:CONSte:IQDC:OFFSet? (Query Only) 1401
FETCh:WLAN:AD:EVM:TRACe? (Query Only) 1401
FETCh:WLAN:AD:SUMMary:AVERage:BURSt:COUNt (Query Only) 1402
FETCh:WLAN:AD:SUMMary:AVERage:BURSt:TOTAl:COUNt (Query Only) 1402
FETCh:WLAN:AD:SUMMary:ESNR (Query Only) 1402
FETCh:WLAN:AD:SUMMary:EVM:DATA:AVERage (Query Only) 1403
FETCh:WLAN:AD:SUMMary:EVM:DATA:AVERage:STATus (Query Only) 1403
FETCh:WLAN:AD:SUMMary:EVM:DATA:MAX (Query Only) 1404
FETCh:WLAN:AD:SUMMary:EVM:PACKet:CEF:EVM (Query Only) 1404
FETCh:WLAN:AD:SUMMary:EVM:PACKet:CEF:PEAK (Query Only) 1405
FETCh:WLAN:AD:SUMMary:EVM:PACKet:CEF:SYMBol (Query Only) 1405
FETCh:WLAN:AD:SUMMary:EVM:PACKet:DATA:EVM (Query Only) 1406
FETCh:WLAN:AD:SUMMary:EVM:PACKet:DATA:PEAK (Query Only) 1406
FETCh:WLAN:AD:SUMMary:EVM:PACKet:DATA:STATus (Query Only) 1407
FETCh:WLAN:AD:SUMMary:EVM:PACKet:DATA:SYMBol (Query Only) 1407
FETCh:WLAN:AD:SUMMary:EVM:PACKet:GUARd:EVM (Query Only) 1408
FETCh:WLAN:AD:SUMMary:EVM:PACKet:GUARd:PEAK (Query Only) 1408
FETCh:WLAN:AD:SUMMary:EVM:PACKet:GUARd:SYMBol (Query Only) 1409
FETCh:WLAN:AD:SUMMary:EVM:PACKet:HEADer:EVM (Query Only) 1409
FETCh:WLAN:AD:SUMMary:EVM:PACKet:HEADer:PEAK (Query Only) 1410
FETCh:WLAN:AD:SUMMary:EVM:PACKet:HEADer:SYMBol (Query Only) 1410

FETCh:WLAN:AD:SUMMary:EVM:PACKet:STF:EVM (Query Only) 1411
FETCh:WLAN:AD:SUMMary:EVM:PACKet:STF:PEAK (Query Only) 1411
FETCh:WLAN:AD:SUMMary:EVM:PACKet:STF:SYMBol (Query Only) 1411
FETCh:WLAN:AD:SUMMary:FERRor:AVERage (Query Only) 1412
FETCh:WLAN:AD:SUMMary:FERRor:AVERage:STATus (Query Only) 1412
FETCh:WLAN:AD:SUMMary:FERRor:HERTz (Query Only) 1413
FETCh:WLAN:AD:SUMMary:FERRor:MAX (Query Only) 1413
FETCh:WLAN:AD:SUMMary:FERRor:PPM (Query Only) 1414
FETCh:WLAN:AD:SUMMary:FERRor:STATus (Query Only) 1414
FETCh:WLAN:AD:SUMMary:IQ:GAIN:IMBALance (Query Only) 1415
FETCh:WLAN:AD:SUMMary:IQ:ORIGin:OFFSet (Query Only) 1415
FETCh:WLAN:AD:SUMMary:IQ:QUADrature:ERROr (Query Only) 1416
FETCh:WLAN:AD:SUMMary:PACKet:AGGRegation (Query Only) 1416
FETCh:WLAN:AD:SUMMary:PACKet:BTR (Query Only) 1417
FETCh:WLAN:AD:SUMMary:PACKet:HCS (Query Only) 1417
FETCh:WLAN:AD:SUMMary:PACKet:LENGth (Query Only) 1418
FETCh:WLAN:AD:SUMMary:PACKet:MCS (Query Only) 1418
FETCh:WLAN:AD:SUMMary:PACKet:PPDU (Query Only) 1418
FETCh:WLAN:AD:SUMMary:PACKet:RSSI (Query Only) 1419
FETCh:WLAN:AD:SUMMary:PACKet:SCRAmbler:INITialization (Query Only) 1419
FETCh:WLAN:AD:SUMMary:PACKet:TAROUND (Query Only) 1420
FETCh:WLAN:AD:SUMMary:PACKet:TLEN (Query Only) 1420
FETCh:WLAN:AD:SUMMary:PACKet:TYPE (Query Only) 1421
FETCh:WLAN:AD:SUMMary:RCPI (Query Only) 1421
FETCh:WLAN:AD:SUMMary:RF:POWer (Query Only) 1422
FETCh:WLAN:AY:CONSte:DATA:EVM:PEAK? (Query Only) 1422
FETCh:WLAN:AY:CONSte:DATA:EVM:PEAK:SYMBol? (Query Only) 1422
FETCh:WLAN:AY:CONSte:DATA:EVM? (Query Only) 1423
FETCh:WLAN:AY:CONSte:FERRor? (Query Only) 1423
FETCh:WLAN:AY:CONSte:FREQuency:ERROr? (Query Only) 1424
FETCh:WLAN:AY:CONSte:IQDC:OFFSet? (Query Only) 1424
FETCh:WLAN:AY:EVM:TRACe? (Query Only) 1424
FETCh:WLAN:AY:SUMMary:AVERage:BURSt:COUNt? (Query Only) 1425
FETCh:WLAN:AY:SUMMary:AVERage:BURSt:TOTAl:COUNt? (Query Only) 1425
FETCh:WLAN:AY:SUMMary:ESNR? (Query Only) 1426
FETCh:WLAN:AY:SUMMary:EVM:DATA:AVERage? (Query Only) 1426
FETCh:WLAN:AY:SUMMary:EVM:DATA:AVERage:STATus? (Query Only) 1427
FETCh:WLAN:AY:SUMMary:EVM:DATA:MAX? (Query Only) 1427
FETCh:WLAN:AY:SUMMary:EVM:DATA:SIGma? (Query Only) 1427
FETCh:WLAN:AY:SUMMary:EVM:PACKet:DATA:EVM? (Query Only) 1428
FETCh:WLAN:AY:SUMMary:EVM:PACKet:DATA:PEAK? (Query Only) 1428
FETCh:WLAN:AY:SUMMary:EVM:PACKet:DATA:STATus? (Query Only) 1429
FETCh:WLAN:AY:SUMMary:EVM:PACKet:DATA:SYMBol? (Query Only) 1429
FETCh:WLAN:AY:SUMMary:EVM:PACKet:EDMG:CEF:EVM? (Query Only) 1430
FETCh:WLAN:AY:SUMMary:EVM:PACKet:EDMG:CEF:PEAK? (Query Only) 1430
FETCh:WLAN:AY:SUMMary:EVM:PACKet:EDMG:CEF:SYMBol? (Query Only) 1431

FETCh:WLAN:AY:SUMMary:EVM:PACKet:EDMG:STF:EVM? (Query Only) 1431
FETCh:WLAN:AY:SUMMary:EVM:PACKet:EDMG:STF:PEAK? (Query Only) 1431
FETCh:WLAN:AY:SUMMary:EVM:PACKet:EDMG:STF:SYMBol? (Query Only) 1432
FETCh:WLAN:AY:SUMMary:EVM:PACKet:GUARd:EVM? (Query Only) 1432
FETCh:WLAN:AY:SUMMary:EVM:PACKet:GUARd:PEAK? (Query Only) 1433
FETCh:WLAN:AY:SUMMary:EVM:PACKet:GUARd:SYMBol? (Query Only) 1433
FETCh:WLAN:AY:SUMMary:EVM:PACKet:HEADer:EVM? (Query Only) 1434
FETCh:WLAN:AY:SUMMary:EVM:PACKet:HEADer:PEAK? (Query Only) 1434
FETCh:WLAN:AY:SUMMary:EVM:PACKet:HEADer:SYMBol? (Query Only) 1435
FETCh:WLAN:AY:SUMMary:EVM:PACKet:PRE1Edmg: A1Header:EVM? (Query 1435
FETCh:WLAN:AY:SUMMary:EVM:PACKet:PRE1Edmg: CEF1:EVM? (Query Only) 1436
FETCh:WLAN:AY:SUMMary:EVM:PACKet:PRE1Edmg: GUARd1:EVM? (Query 1436
FETCh:WLAN:AY:SUMMary:EVM:PACKet:PRE1Edmg :HEADer1:EVM? (Query 1436
FETCh:WLAN:AY:SUMMary:EVM:PACKet:PRE1Edmg:STF1:EVM? (Query Only) 1437
FETCh:WLAN:AY:SUMMary:EVM:PACKet:PRE2Edmg: A2Header:EVM? (Query 1437
FETCh:WLAN:AY:SUMMary:EVM:PACKet:PRE2Edmg:CEF2:EVM? (Query Only) 1438
FETCh:WLAN:AY:SUMMary:EVM:PACKet:PRE2Edmg: GUARd2:EVM? (Query 1438
FETCh:WLAN:AY:SUMMary:EVM:PACKet:PRE2Edmg: HEADer2:EVM? (Query 1439
FETCh:WLAN:AY:SUMMary:EVM:PACKet:PRE2Edmg: STF2:EVM? (Query Only) 1439
FETCh:WLAN:AY:SUMMary:FERRor:AVERage? (Query Only) 1440
FETCh:WLAN:AY:SUMMary:FERRor:AVERage:STATus? (Query Only) 1440
FETCh:WLAN:AY:SUMMary:FERRor:HERTz? (Query Only) 1440
FETCh:WLAN:AY:SUMMary:FERRor:MAX? (Query Only) 1441
FETCh:WLAN:AY:SUMMary:FERRor:PPM? (Query Only) 1441
FETCh:WLAN:AY:SUMMary:FERRor:STATus? (Query Only) 1442
FETCh:WLAN:AY:SUMMary:IQ:GAIN:IMBALance? (Query Only) 1442
FETCh:WLAN:AY:SUMMary:IQ:ORIGin:OFFSet? (Query Only) 1442
FETCh:WLAN:AY:SUMMary:IQ:QUADrature:ERROr? (Query Only) 1443
FETCh:WLAN:AY:SUMMary:PACKet:AGGRegation? (Query Only) 1443
FETCh:WLAN:AY:SUMMary:PACKet:BEAMformed? (Query Only) 1444
FETCh:WLAN:AY:SUMMary:PACKet:BWBitmap? (Query Only) 1444
FETCh:WLAN:AY:SUMMary:PACKet:COMPbw? (Query Only) 1444
FETCh:WLAN:AY:SUMMary:PACKet:CRC? (Query Only) 1445
FETCh:WLAN:AY:SUMMary:PACKet:EDMGmcs? (Query Only) 1445
FETCh:WLAN:AY:SUMMary:PACKet:LDPCcwlen? (Query Only) 1446
FETCh:WLAN:AY:SUMMary:PACKet:PRIMarychannel? (Query Only) 1446
FETCh:WLAN:AY:SUMMary:PACKet:PSDUlen? (Query Only) 1446
FETCh:WLAN:AY:SUMMary:PACKet:RSSI? (Query Only) 1447
FETCh:WLAN:AY:SUMMary:PACKet:SCRAmbler:INITialization? (Query Only) 1447
FETCh:WLAN:AY:SUMMary:PACKet:SPATialstreams? (Query Only) 1448
FETCh:WLAN:AY:SUMMary:PACKet:STBCapplied? (Query Only) 1448
FETCh:WLAN:AY:SUMMary:PACKet:TRNLen ? (Query Only) 1448
FETCh:WLAN:AY:SUMMary:RCPI? (Query Only) 1449
FETCh:WLAN:AY:SUMMary:RF:POWer? (Query Only) 1449
FETCh:WLAN:AY:SUMMary:SIG:QUAlity? (Query Only) 1449
FETCh:WLAN:CONSte:MAGNitude? (Query Only) 1450

FETCh:WLAN:CONSte:PHASe? (Query Only) 1450
FETCh:WLAN:CONSte:TYPE? (Query Only) 1451
FETCh:WLAN:CONSte:VALue? (Query Only) 1452
FETCh:WLAN:CRESponse:MAGNitude? (Query Only) 1452
FETCh:WLAN:CRESponse:PHASe? (Query Only) 1453
FETCh:WLAN:EVM:TRACe<x>? (Query Only) 1453
FETCh:WLAN:FLATness:PASS? (Query Only) 1454
FETCh:WLAN:FLATness:RESult? (Query Only) 1454
FETCh:WLAN:FLATness:TRACe<x>? (Query Only) 1455
FETCh:WLAN:MERRor:TRACe<x>? (Query Only) 1455
FETCh:WLAN:PERRor:TRACe<x>? (Query Only) 1456
FETCh:WLAN:PVTime:BURSt:POWer? (Query Only) 1457
FETCh:WLAN:PVTime:BURSt:WIDTh? (Query Only) 1457
FETCh:WLAN:PVTime:FALL:TIME? (Query Only) 1458
FETCh:WLAN:PVTime:FALL:TIME:STARt? (Query Only) 1458
FETCh:WLAN:PVTime:FALL:TIME:STOP? (Query Only) 1458
FETCh:WLAN:PVTime:FTIMe? (Query Only) 1459
FETCh:WLAN:PVTime:RISE:TIME? (Query Only) 1459
FETCh:WLAN:PVTime:RISE:TIME:STARt? (Query Only) 1460
FETCh:WLAN:PVTime:RISE:TIME:STOP? (Query Only) 1460
FETCh:WLAN:PVTime:RTIMe? (Query Only) 1460
FETCh:WLAN:PVTime:TRACe:X? (Query Only) 1461
FETCh:WLAN:PVTime:TRACe:XY? (Query Only) 1461
FETCh:WLAN:PVTime:TRACe[:Y]? (Query Only) 1462
FETCh:WLAN:STABle:VALUe? (Query Only) 1462
FETCh:WLAN:SUMMary:BURSt:INDex? (Query Only) 1463
FETCh:WLAN:SUMMary:BURSt:POWer? (Query Only) 1463
FETCh:WLAN:SUMMary:BURSt:POWer:CFACtor? (Query Only) 1464
FETCh:WLAN:SUMMary:CERRor? (Query Only) 1464
FETCh:WLAN:SUMMary:CPE? (Query Only) 1465
FETCh:WLAN:SUMMary:EVM:PEAK:ALL? (Query Only) 1465
FETCh:WLAN:SUMMary:EVM:PEAK:CHIP:ALL? (Query Only) 1465
FETCh:WLAN:SUMMary:EVM:PEAK:CHIP:ONEKchips? (Query Only) 1466
FETCh:WLAN:SUMMary:EVM:PEAK:DATA? (Query Only) 1466
FETCh:WLAN:SUMMary:EVM:PEAK:ONEKchips? (Query Only) 1467
FETCh:WLAN:SUMMary:EVM:PEAK:PILots? (Query Only) 1467
FETCh:WLAN:SUMMary:EVM:PEAK:SCARrier:ALL? (Query Only) 1467
FETCh:WLAN:SUMMary:EVM:PEAK:SCARrier:DATA? (Query Only) 1468
FETCh:WLAN:SUMMary:EVM:PEAK:SCARrier:PILots? (Query Only) 1468
FETCh:WLAN:SUMMary:EVM:PEAK:SYMBol:ALL? (Query Only) 1469
FETCh:WLAN:SUMMary:EVM:PEAK:SYMBol:DATA? (Query Only) 1469
FETCh:WLAN:SUMMary:EVM:PEAK:SYMBol:PILots? (Query Only) 1469
FETCh:WLAN:SUMMary:EVM:RMS:ALL? (Query Only) 1470
FETCh:WLAN:SUMMary:EVM:RMS:AVERage? (Query Only) 1470
FETCh:WLAN:SUMMary:EVM:RMS:DATA? (Query Only) 1471
FETCh:WLAN:SUMMary:EVM:RMS:MAXimum? (Query Only) 1471

FETCh:WLAN:SUMMary:EVM:RMS:ONEKchips? (Query Only) 1471
FETCh:WLAN:SUMMary:EVM:RMS:PILots? (Query Only) 1472
FETCh:WLAN:SUMMary:FERRor? (Query Only) 1472
FETCh:WLAN:SUMMary:HEADer? (Query Only) 1473
FETCh:WLAN:SUMMary:HTSig? (Query Only) 1473
FETCh:WLAN:SUMMary:IQOFfset? (Query Only) 1473
FETCh:WLAN:SUMMary:LSIG? (Query Only) 1474
FETCh:WLAN:SUMMary:PACKet? (Query Only) 1474
FETCh:WLAN:SUMMary:PACKet:DMODulation? (Query Only) 1475
FETCh:WLAN:SUMMary:PACKet:FORMat? (Query Only) 1475
FETCh:WLAN:SUMMary:PACKet:GUARd? (Query Only) 1475
FETCh:WLAN:SUMMary:SIG? (Query Only) 1476
FETCh:WLAN:SUMMary:VHTSig:A? (Query Only) 1476
FETCh:WLAN:SUMMary:VHTSig:B? (Query Only) 1477
*IDN? (Query Only) 1477
INITiate:CONTinuous 1478
INITiate[:IMMediate] (No Query Form) 1478
INITiate:RESume (No Query Form) 1479
Input commands 1480
INPut:CORRection:EXTernal:EDIT<x>:INTerpolation 1480
INPut:CORRection:EXTernal:EDIT<x>:LABel 1480
INPut:CORRection:EXTernal:EDIT<x>:NEW 1481
INPut:CORRection:EXTernal:EDIT<x>:STATe 1481
INPut:CORRection:EXTernal:GAIN[:MAGNitude] 1482
INPut:CORRection:EXTernal:GAIN:STATe 1482
INPut:CORRection:EXTernal:TYPE 1483
INPut:{MLEVel|RLEVel} 1484
INPut[:RF]:ATTenuation 1484
INPut[:RF]:ATTenuation:AUTO 1484
INPut[:RF]:ATTenuation:MONitor:STATe 1485
INPut[:RF]:GAIN:STATe 1485
INPut:TYPE 1486
Memory commands 1486
MMEMory:{AM|FM|PM}:LOAD:TRACe (No Query Form) 1486
MMEMory:{AM|FM|PM}:SHOW:TRACe<x> 1487
MMEMory:{AM|FM|PM}:STORe:TRACe (No Query Form) 1487
MMEMory:AUDio:LOAD:FILTer (No Query Form) 1487
MMEMory:AVTime[:MEASview<y>]:LOAD:TRACe<x> (No Query Form) 1488
MMEMory:AVTime[:MEASview<y>]:SHOW:TRACe<x> 1488
MMEMory:AVTime[:MEASview<y>]:STORe:PLOT 1489
MMEMory:AVTime[:MEASview<y>]:STORe:PLOT:DONe 1489
MMEMory:AVTime[:MEASview<y>]:STORe:TRACe<x> (No Query Form) 1490
MMEMory:CALibration:LOAD:CORRection:EXTernal:EDIT<x> (No Query Form) 1490
MMEMory:CALibration:STORe:CORRection:EXTernal:EDIT<x> (No Query Form) 1491
MMEMory:CCDF:LOAD:TRACe<x> (No Query Form) 1491
MMEMory:CCDF:SHOW:TRACe<x> 1492

MMEMory:CCDF:STORe:TRACe<x> (No Query Form) 1492
MMEMory:DDEMod[:SOURce<x>]:LOAD:FILTer:MEASurement:UOTHer (No query form) 1493
MMEMory:DDEMod[:SOURce<x>]:LOAD:FILTer:MEASurement: USER<y> (No query form) 1493
MMEMory:DDEMod[:SOURce<x>]:LOAD:FILTer:REFerence:UOTHer (No query form) 1494
MMEMory:DDEMod[:SOURce<x>]:LOAD:FILTer:REFerence:USER<y> (No query form) 1494
MMEMory:DDEMod[:SOURce<x>]:LOAD:SYMBol:MAP (No query form) 1495
MMEMory:DPX:LOAD:TRACe<x> (No Query Form) 1495
MMEMory:DPX:SHOW:TRACe<x> 1495
MMEMory:DPX:STORe:TRACe<x> 1496
MMEMory:DPX:TDM:FREQuency:TXBer:LOAD:PATTern (No Query Form) 1496
MMEMory:EMC:ACCEssory<x>:LOAD:TABLe (No Query Form) 1497
MMEMory:EMC:ACCEssory<x>:STORe:TABLe (No Query Form) 1497
MMEMory:EMC:AMBIent:TRACe:FILE (No Query Form) 1498
MMEMory:EMC:AMBIent:TRACe:SHOW (No Query Form) 1498
MMEMory:EMC:AMBIent:TRACe:STORe (No Query Form) 1499
MMEMory:EMC:MATH:TRACe:FILE (No Query Form) 1499
MMEMory:EMC:MATH:TRACe:SHOW 1500
MMEMory:EMC:MATH:TRACe:STORe (No Query Form) 1500
MMEMory:EMC:RANGe:LOAD:TABLe (No Query Form) 1501
MMEMory:EMC:RANGe:STORe:TABLe (No Query Form) 1501
MMEMory:EMC:RECAll:TRACe<x>:FILE (No Query Form) 1502
MMEMory:EMC:RECAll:TRACe<x>:SHOW (No Query Form) 1502
MMEMory:EMC:REPOrt:GENerate:FINAl (No Query Form) 1503
MMEMory:EMC:REPOrt:HISTory:DELete:ALL (No Query Form) 1503
MMEMory:EMC:REPOrt:SAVE:RUN (No Query Form) 1503
MMEMory:EMC:SCAN:TRACe<x>:STORe (No Query Form) 1504
MMEMory:EMC:SPOT:TRACe<x>:STORe (No Query Form) 1504
[MMEMory:{FSETtling|PSETtling}:LOAD:TRACe<x> 1505
[MMEMory:{FSETtling|PSETtling}:SHOW:TRACe<x> 1505
[MMEMory:{FSETtling|PSETtling}:STORe:TRACe<x> 1506
MMEMory:FVTime[:MEASview<y>]:LOAD:TRACe (No Query Form) 1506
MMEMory:FVTIME[:MEASview<y>]:SHOW:TRACe<x> 1507
MMEMory:FVTime[:MEASview<y>]:STORe:TRACe (No Query Form) 1507
MMEMory:IQVTime[:MEASVIEW<y>]:LOAD:TRACe:I (No Query Form) 1508
MMEMory:IQVTime[:MEASVIEW<y>]:LOAD:TRACe:Q (No Query Form) 1508
MMEMory:IQVTIME[:MEASVIEW<y>]:SHOW:TRACe:I 1509
MMEMory:IQVTIME[:MEASVIEW<y>]:SHOW:TRACe<x>:Q 1509
MMEMory:IQVTime[:MEASVIEW<y>]:STORe:TRACe:I (No Query Form) 1510
MMEMory:IQVTime[:MEASVIEW<y>]:STORe:TRACe:Q (No Query Form) 1510
MMEMory:LOAD:IQ (No Query Form) 1511
MMEMory:LOAD:STATe (No Query Form) 1511
MMEMory:LOAD:TRACe (No Query Form) 1511
MMEMory:LTE:ACLR:LOAD:TABLe (No Query Form) 1513
MMEMory:LTE:ACLR:STORe:TABLe (No Query Form) 1514
MMEMory:MAPit:LOAD (No Query Form) 1514
MMEMory:MAPit:STORe (No Query Form) 1515

MMEMory:MAPit:STORe:ALL (No Query Form) . . . 1515
MMEMory:NOISe:GAIN:LOAD:TRACe (No Query Form) . . . 1515
MMEMory:NOISe:GAIN:SHOW:TRACe . . . 1516
MMEMory:NOISe:GAIN:STORe:TRACe (No Query Form) . . . 1516
MMEMory:Noise:LOAD:ENRCALibration (No Query Form) . . . 1517
MMEMory:Noise:LOAD:FREQTABle (No Query Form) . . . 1517
MMEMory:Noise:STORe:ENRCALibration (No Query Form) . . . 1518
MMEMory:Noise:STORe:FREQTABle (No Query Form) . . . 1518
MMEMory:OFDM:CONSte:LOAD:TRACe (No Query Form) . . . 1518
MMEMory:OFDM:CONSTe:SHOW:TRACe . . . 1519
MMEMory:OFDM:CONSte:STORe:TRACe (No Query Form) . . . 1519
MMEMory:PHVTime[:MEASVIEW<y>]:LOAD:TRACe (No Query Form) . . . 1520
MMEMory:PHVTime[:MEASview<y>]:SHOW:TRACe . . . 1520
MMEMory:PHVTime[:MEASVIEW<y>]:STORe:TRACe (No Query Form) . . . 1521
MMEMory:PNOise:LOAD:TRACe<x> (No Query Form) . . . 1521
MMEMory:PNOise:SHOW:TRACe<x> . . . 1521
MMEMory:PNOise:STORe:TRACe<x> (No Query Form) . . . 1522
MMEMory:SEM:LOAD:TABLe (No Query Form) . . . 1522
MMEMory:SEM:STORe:TABLe (No Query Form) . . . 1523
MMEMory:SGRam[:MEASview<y>]:LOAD:TRACe (No Query Form) . . . 1523
MMEMory:SGRam[:MEASview<y>]:SHOW:TRACe . . . 1524
MMEMory:SGRam[:MEASview<y>]:STORe:TRACe (No Query Form) . . . 1524
MMEMory:SGRam[:MEASview<y>]:STORe:PLOT . . . 1525
MMEMory:SGRam[:MEASview<y>]:STORe:PLOT:DONe (Query only) . . . 1525
MMEMory:SPECtrum:MEASview<y>:LOAD:TRACe (No Query Form) . . . 1526
MMEMory:SPECtrum[:MEASview<y>]:SHOW:TRACe<x> . . . 1526
MMEMory:SPECtrum[:MEASview<y>]:STORe:PLOT (No Query Form) . . . 1527
MMEMory:SPECtrum[:MEASview<y>]:STORe:PLOT:DONe (Query Only) . . . 1527
MMEMory:SPECtrum[:MEASview<y>]:STORe:TRACe (No Query Form) . . . 1528
MMEMory:SPURious:LOAD:TABLe (No Query Form) . . . 1528
MMEMory:SPURious:RECAll:TRACe (No Query Form) . . . 1529
MMEMory:SPURious:SHOW:TRACe (No Query Form) . . . 1529
MMEMory:SPURious:STORe:TABLe (No Query Form) . . . 1530
MMEMory:SPURious:STORe:TRACe (No Query Form) . . . 1530
MMEMory:STORe:IQ (No Query Form) . . . 1531
MMEMory:STORe:IQ:CSV (No Query Form) . . . 1531
MMEMory:STORe:IQ:MAT (No Query Form) . . . 1531
MMEMory:STORe:IQ:SELEct:DATA . . . 1532
MMEMory:STORe:IQ:SELEct:FRAMes . . . 1532
MMEMory:STORe:IQ:SELEct:LENGth . . . 1533
MMEMory:STORe:MSTate (No Query Form) . . . 1533
MMEMory:STORe:RESults (No Query Form) . . . 1533
MMEMory:STORe:SCReen (No Query Form) . . . 1534
MMEMory:STORe:STATe (No Query Form) . . . 1534
MMEMory:STORe:TRACe (No Query Form) . . . 1535
MMEMory:TOVerview[:MEASview<y>]:LOAD:TRACe1 (No Query Form) . . . 1535

MMEMory:TOVerview[:MEASview<y>]:SHOW:TRACe1 1536
MMEMory:TOVerview[:MEASview<y>]:STORe:TRACe1 (No Query Form) 1536
MMEMory:TXGain:LOAD:TRACe<x> (No Query Form) 1537
MMEMory:TXGain:SHOW:TRACe<x> 1537
MMEMory:TXGain:STORe:TRACe<x> (No Query Form) 1538
MMEMory:WLAN:AD:SUMMary:LIMIt:TABLe:LOAD (No Query Form) 1538
MMEMory:WLAN:AD:SUMMary:LIMIt:TABLe:SAVE (No Query Form) 1538
MMEMory:WLAN:AY:SUMMary:LIMIt:TABLe:LOAD (No Query Form) 1539
MMEMory:WLAN:AY:SUMMary:LIMIt:TABLe:SAVE (No Query Form) 1539
MMEMory:WLAN:CRESponse:LOAD:TRACe<x> (No Query Form) 1540
MMEMory:WLAN:CRESponse:SHOW:TRACe<x> 1540
MMEMory:WLAN:CRESponse:STORe:TRACe<x> (No Query Form) 1541
TEST 1541
*OPC 1542
*OPT? (Query Only) 1542
Output commands 1544
OUTPut:IF:{BANDwidth|BWIDth} 1544
OUTPut:IF[:STATe] 1544
OUTPut:IQ[:STATe] 1545
OUTPut:NOISe[:STATe] 1545
OUTPut:IQ:SOURce 1545
OUTPut:TGen:BRIDge:MODE 1546
OUTPut:TGen:ENABle 1546
OUTPut:TGen:FREQuency 1547
OUTPut:TGen:INSTalled? (Query Only) 1547
OUTPut:TGen:LEVel 1548
OUTPut:ZS:{BANDwidth|BWIDth} 1548
OUTPut:ZS:{BANDwidth|BWIDth}:AUTO 1548
OUTPut:ZS:{BANDwidth|BWIDth}:STATe 1549
OUTPut:ZS:DELay? (Query Only) 1549
OUTPut:ZS:FILTer:STATe 1550
OUTPut:ZS[:STATe] 1550
OUTPut:ZS:VOLTage 1550
OUTPut:ZS:VOLTage:AUTO 1551
Read commands 1551
READ:ACPower? (Query Only) 1551
READ:ACPower:CHANnel:POWer? (Query Only) 1552
READ:ACPower:SPECtrum? (Query Only) 1552
READ:{AM|FM|PM}? (Query Only) 1553
READ:AM:AMINdex? (Query Only) 1554
READ:AM:AMNegative? (Query Only) 1554
READ:AM:AMPositive? (Query Only) 1554
READ:AM:RESult? (Query Only) 1555
READ:AUDio:FERRor? (Query Only) 1555
READ:AUDio:FREQuency? (Query Only) 1556
READ:AUDio:HARMonic:COUNt? (Query Only) 1556

READ:AUDio:HARMonic<x>:AMPLitude? (Query Only) 1557
READ:AUDio:HARMonic<x>:FREQuency? (Query Only) 1557
READ:AUDio:HNOise? (Query Only) 1558
READ:AUDio:HNREFerence? (Query Only) 1559
READ:AUDio:HPTPeak? (Query Only) 1559
READ:AUDio:MODDist? (Query Only) 1560
READ:AUDio:NHARmonic:COUNt? (Query Only) 1560
READ:AUDio:NHARmonic<x>:AMPLitude? (Query Only) 1561
READ:AUDio:NHARmonic<x>:FREQuency? (Query Only) 1561
READ:AUDio:PNEGative? (Query Only) 1562
READ:AUDio:POWer? (Query Only) 1562
READ:AUDio:PPOSitive? (Query Only) 1563
READ:AUDio:RMS? (Query Only) 1563
READ:AUDio:SINad? (Query Only) 1564
READ:AUDio:SNOise? (Query Only) 1564
READ:AUDio:SPECtrum:TRACe<x>? (Query Only) 1565
READ:AUDio:THDist? (Query Only) 1565
READ:AUDio:THDist:DB? (Query Only) 1566
READ:AUDio:TNHDist? (Query Only) 1566
READ:AUDio:TNHDist:DB? (Query Only) 1567
READ:AVTime[:MEASview<y>]:AVERage? (Query Only) 1567
READ:AVTime[:MEASview<y>]:{FIRSt|SECond|THIRd|FOURth}? (Query Only) 1568
READ:AVTime[:MEASview<y>]:MAXimum? (Query Only) 1569
READ:AVTime[:MEASview<y>]:MAXLocation? (Query Only) 1569
READ:AVTime[:MEASview<y>]:MINimum? (Query Only) 1570
READ:AVTime[:MEASview<y>]:MINLocation? (Query Only) 1570
READ:AVTime[:MEASview<y>]:RESult? (Query Only) 1571
READ:AVTime[:MEASview<y>]:AVERage:{FIRSt|SECond|THIRd|FOURth}? 1572
READ:AVTime[:MEASview<y>]:MAXimum:{FIRSt|SECond|THIRd|FOURth}? 1572
READ:AVTime[:MEASview<y>]:MAXLocation:{FIRSt|SECond|THIRd|FOURth}? 1573
READ:AVTime[:MEASview<y>]:MINimum:{FIRSt|SECond|THIRd|FOURth}? 1574
READ:AVTime[:MEASview<y>]:MINLocation:{FIRSt|SECond|THIRd|FOURth}? 1574
READ:AVTime[:MEASview<y>]:OGRAm? (Query only) 1575
READ:AVTime[:MEASview<y>]:RESult:{FIRSt|SECond|THIRd|FOURth}? 1575
READ:AVTime[:MEASview<y>]:TRACe:BITMap? (Query Only) 1576
READ:BIBEmissions:FTX? (Query Only) 1577
READ:BIBEmissions:POWer? (Query Only) 1577
READ:BIBEmissions:RESUlts:STATus? (Query Only) 1578
READ:BLUEtooth:CONSte:FERRor? (Query Only) 1578
READ:BLUEtooth:CONSte:TRACe? (Query Only) 1579
READ:BLUEtooth:EDIagram:FDEViation? (Query Only) 1579
READ:BLUEtooth:EDIagram:FERRor? (Query Only) 1580
READ:BLUEtooth:EDIagram:HORiz:SCALe? (Query Only) 1580
READ:BLUEtooth:EDIagram:I? (Query Only) 1581
READ:BLUEtooth:EDIagram:Q? (Query Only) 1581
READ:BLUEtooth:FDVTime:TRACe? (Query Only) 1582

READ:BLUEtooth:FREQuency:ERROr? (Query Only) 1582
READ:BLUEtooth:FREQuency:ERROr:TYPE 1583
READ:BLUEtooth:FREQuency:OFFSet:DRIFt:F1FZero? (Query Only) 1583
READ:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFN5? (Query Only) 1584
READ:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFN5:INTErval? (Query Only) 1584
READ:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFZERO? (Query Only) 1585
READ:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFZERO:INTErval? (Query Only) 1585
READ:BLUEtooth:FREQuency:OFFSet:MAX? (Query Only) 1586
READ:BLUEtooth:FREQuency:OFFSet:MAX:INTErval? (Query Only) 1586
READ:BLUEtooth:FREQuency:OFFSet:PREAmble? (Query Only) 1587
READ:BLUEtooth:MODUlation:CHARacteristics:AVErage:F? (Query Only) 1587
READ:BLUEtooth:MODUlation:CHARacteristics:AVErage:F2F1ratio? (Query Only) 1588
READ:BLUEtooth:MODUlation:CHARacteristics:AVErage:FONE? (Query Only) 1588
READ:BLUEtooth:MODUlation:CHARacteristics:AVErage:FTWO? (Query Only) 1589
READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:COUNt? (Query Only) 1589
READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:F1FZero? (Query Only) 1590
READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:F1FZero:STATus? (Query Only) 1590
READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFN5? (Query Only) 1591
READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFN5:STATus? (Query Only) 1591
READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFZero? (Query Only) 1592
READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFZero: STATus? (Query Only) 1592
READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded: CODIng:SCHEme (Query Only) 1593
READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded: F0FThree (Query Only) 1593
READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded: F0FThree:STATus (Query Only) 1594
READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded: F1FFour (Query Only) 1594
READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded: F1FFour:STATus (Query Only) 1595
READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded: FNFN3 (Query Only) 1595
READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded: FNFN3:STATus (Query Only) 1596
READ:BLUEtooth:SUMMary:FREQuency:OFFSet:MAX? (Query Only) 1596
READ:BLUEtooth:SUMMary:FREQuency:OFFSet:MAX:STATus? (Query Only) 1597
READ:BLUEtooth:SUMMary:FREQuency:OFFSet:PREAmble? (Query Only) 1597
READ:BLUEtooth:SUMMary:FREQuency:OFFSet:PREAmble:STATus? (Query Only) 1598
READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics: AVErage:DELTa:F:MAX:PERCentage:COUNt? (Query Only) 1598
READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics: AVErage:DELTa:FONE:MAX:PERCentage:COUNt? (Query Only) 1599
READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics: AVErage:DELTa:FTWO:MAX:PERCentage? (Query Only) 1599
READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics: AVErage:DELTa:FTWO:MAX:PERCentage:COUNt? (Query Only) 1600
READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics: AVErage:DELTa:FTWO:MAX:PERCentage:STATus? (Query Only) 1600
READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics: AVErage:F2F1ratio? (Query Only) 1601
READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics: AVErage:F2F1ratio:STATus? (Query Only) 1601
READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics: AVErage:F? (Query Only) 1602
READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics: AVErage:FONE? (Query Only) 1602
READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics: AVErage:FONE:STATus? (Query Only) 1603
READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics: AVErage:FTWO? (Query Only) 1603

READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:INDex (Query Only) 1604
READ:BLUEtooth:SUMMary:OUTPut:POWEr:AVErage? (Query Only) 1604
READ:BLUEtooth:SUMMary:OUTPut:POWEr:AVErage:STATus? (Query Only) 1605
READ:BLUEtooth:SUMMary:OUTPut:POWEr:COUNt? (Query Only) 1605
READ:BLUEtooth:SUMMary:OUTPut:POWEr:EDR:COUNt? (Query Only) 1606
READ:BLUEtooth:SUMMary:OUTPut:POWEr:EDR:DPSK? (Query Only) 1606
READ:BLUEtooth:SUMMary:OUTPut:POWEr:EDR:GPSK? (Query Only) 1607
READ:BLUEtooth:SUMMary:OUTPut:POWEr:PEAK? (Query Only) 1607
READ:BLUEtooth:SUMMary:OUTPut:POWEr:PEAK:STATus? (Query Only) 1608
READ:BLUEtooth:SUMMary:PACKet:BDR:CRC? (Query Only) 1608
READ:BLUEtooth:SUMMary:PACKet:BDR:HEADer:ARQN? (Query Only) 1609
READ:BLUEtooth:SUMMary:PACKet:BDR:HEADer:FLOW? (Query Only) 1609
READ:BLUEtooth:SUMMary:PACKet:BDR:HEADer:HEC? (Query Only) 1610
READ:BLUEtooth:SUMMary:PACKet:BDR:HEADer:LT:ADDRess? (Query Only) 1610
READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:SN (Query Only) 1611
READ:BLUEtooth:SUMMary:PACKet:BDR:HEADer:SEQN? (Query Only) 1611
READ:BLUEtooth:SUMMary:PACKet:BDR:HEADer:TYPE? (Query Only) 1612
READ:BLUEtooth:SUMMary:PACKet:BDR:PAYLoad:LENgth? (Query Only) 1612
READ:BLUEtooth:SUMMary:PACKet:BDR:PREAmble? (Query Only) 1613
READ:BLUEtooth:SUMMary:PACKet:BDR:SYNC:WORD? (Query Only) 1613
READ:BLUEtooth:SUMMary:PACKet:BDR:TYPE? (Query Only) 1613
READ:BLUEtooth:SUMMary:PACKet:BLE:ACCEss:ADDRess? (Query Only) 1614
READ:BLUEtooth:SUMMary:PACKet:BLE:CRC? (Query Only) 1614
READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer: CHANnel:SELect (Query Only) 1615
READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:LENGth? (Query Only) 1615
READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:LLID (Query Only) 1616
READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:MD (Query Only) 1616
READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:NESN (Query Only) 1617
READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:RX:ADDRess? (Query Only) 1617
READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:TX:ADDRess? (Query Only) 1618
READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:TYPE? (Query Only) 1618
READ:BLUEtooth:SUMMary:PACKet:BLE:PREAmble? (Query Only) 1619
READ:BLUEtooth:SUMMary:PACKet:BLE:TYPE? (Query Only) 1619
READ:CCDF? (Query Only) 1620
READ:CCDF:{FIRSt|SECond|THIRd}:X? (Query Only) 1621
READ:CCDF:{FIRSt|SECond|THIRd}:XY? (Query Only) 1621
READ:CCDF:{FIRSt|SECond|THIRd}[:Y]? (Query Only) 1622
READ:CONSte[:MEASview<y>]:FERRor? (Query only) 1623
READ:CONSte[:MEASview<y>]:RESults? (Query only) 1623
READ:CONSte[:MEASview<y>]:TRACe? (Query only) 1624
READ:DDEMod[:MEASview<y>]:STABle? (Query only) 1625
READ:DIQVtime[:MEASview<y>]:FERRor? (Query only) 1626
READ:DIQVtime[:MEASview<y>]:I? (Query only) 1626
READ:DIQVtime[:MEASview<y>]:Q? (Query only) 1627
READ:DPX:DDENsity? (Query Only) 1627
READ:DPX:DGRam:LINE:COUNt? (Query Only) 1628

READ:DPX:DGRam:TIME[:SCALe]:OFFSet? (Query Only) 1628
READ:DPX:RESults:TRACe<x>? (Query Only) 1629
READ:DPX:TRACe:AVERage? (Query Only) 1629
READ:DPX:TRACe:BITMap? (Query Only) 1630
READ:DPX:TRACe:MATH? (Query Only) 1630
READ:DPX:TRACe:MAXimum? (Query Only) 1631
READ:DPX:TRACe:MINimum? (Query Only) 1632
READ:EDIagram[:MEASview<y>]:FDEViation? (Query only) 1632
READ:EDIagram[:MEASview<y>]:FERRor? (Query only) 1633
READ:EDIagram[:MEASview<y>]:I? (Query only) 1633
READ:EDIagram[:MEASview<y>]:Q? (Query only) 1634
READ:EMC:COUNt? (Query Only) 1635
READ:EMC:EMISsion:RESults:STATus? (Query Only) 1635
READ:EMC:SPOT<x>:AMBIent:AMPLitude? (Query Only) 1635
READ:EMC:SPOT<x>:AMBIent:DELTa? (Query Only) 1636
READ:EMC:SPOT<x>:AMBIent:PASS? (Query Only) 1636
READ:EMC:SPOT<x>:FREQuency? (Query Only) 1637
READ:EMC:SPOT<x>:PASS? (Query Only) 1637
READ:EMC:SPOT<x>:RANGe? (Query Only) 1638
READ:EMC:SPOT<x>:RECAll:AMPLitude? (Query Only) 1638
READ:EMC:SPOT<x>:RECAll:DELTa? (Query Only) 1639
READ:EMC:SPOT<x>:RECAll:PASS? (Query Only) 1639
READ:EMC:SPOT<x>:SCAN<y>:AMPLitude? (Query Only) 1640
READ:EMC:SPOT<x>:SCAN<y>:DELTa? (Query Only) 1641
READ:EMC:SPOT<x>:SCAN<y>:PASS? (Query Only) 1641
READ:EMC:SPOT<x>:SPOT<y>:AMPLitude? (Query Only) 1642
READ:EMC:SPOT<x>:SPOT<y>:DELTa? (Query Only) 1642
READ:EMC:SPOT<x>:SPOT<y>:PASS? (Query Only) 1643
READ:EVM[:MEASview<y>]:FERRor? (Query only) 1643
READ:EVM[:MEASview<y>]:PEAK? (Query only) 1644
READ:EVM[:MEASview<y>]:PINDex? (Query only) 1644
READ:EVM[:MEASview<y>]:RMS? (Query only) 1645
READ:EVM[:MEASview<y>]:TRACe? (Query only) 1646
READ:FDVTime[:MEASview<y>]:FERRor? (Query only) 1646
READ:FDVTime[:MEASview<y>]:TRACe? (Query only) 1647
READ:{FM|PM}:FERRor? (Query only) 1647
READ:FM:PHALf? (Query Only) 1648
READ:FM:PNEGative? (Query Only) 1648
READ:FM:PPOSitive? (Query Only) 1649
READ:FM:PTPeak? (Query Only) 1649
READ:FM:RESult? (Query Only) 1650
READ:FM:RMS? (Query Only) 1650
READ:{FSETtling|PSETtling}:FTTime? (Query Only) 1651
READ:{FSETtling|PSETtling}:MASK[:PASS]? (Query Only) 1651
READ:{FSETtling|PSETtling}:SETTled:FREQuency? (Query Only) 1652
READ:{FSETtling|PSETtling}:SETTled[:PASS]? (Query Only) 1652

READ:{FSETtling|PSETtling}:SETTled:TIME? (Query Only) 1653
READ:{FSETtling|PSETtling}:SLMSd[:PASS]? (Query Only) 1653
READ:{FSETtling|PSETtling}:STARt:TIME? (Query Only) 1653
READ:{FSETtling|PSETtling}:TIME? (Query Only) 1654
READ:{FSETtling|PSETtling}:TRACe<x>:X? (Query Only) 1654
READ:{FSETtling|PSETtling}:TRACe<x>:XY? (Query Only) 1655
READ:{FSETtling|PSETtling}:TRACe<x>[:Y]? (Query Only) 1656
READ:{FSETtling|PSETtling}:TRIGger:TIME? (Query Only) 1656
READ:{FSETtling|PSETtling}:VALue? (Query Only) 1657
READ:FSETtling:ERRor? (Query Only) 1657
READ:FVTime[:MEASview<y>]? (Query Only) 1658
READ:FVTime[:MEASview<y>]:MAXimum? (Query Only) 1658
READ:FVTime[:MEASview<y>]:MAXLocation? (Query Only) 1659
READ:FVTime[:MEASview<y>]:MINimum? (Query Only) 1659
READ:FVTime[:MEASview<y>]:MINLocation? (Query Only) 1660
READ:FVTime[:MEASview<y>]:OGRAm? (Query only) 1660
READ:FVTime[:MEASview<y>]:RESult? (Query Only) 1661
READ:FVTime[:MEASview<y>]:TRACe:BITMap? (Query Only) 1662
READ:IQVTime[:MEASview<y>]:I? (Query Only) 1662
READ:IQVTime[:MEASview<y>]:MAXimum? (Query Only) 1663
READ:IQVTime[:MEASview<y>]:MAXLocation? (Query Only) 1663
READ:IQVTime[:MEASview<y>]:MINimum? (Query Only) 1664
READ:IQVTime[:MEASview<y>]:MINLocation? (Query Only) 1664
READ:IQVTime[:MEASview<y>]:Q? (Query Only) 1665
READ:IQVTime[:MEASview<y>]:RESult? (Query Only) 1665
READ:LTE:ACLR:MHITs:COUNt? (Query Only) 1666
READ:LTE:ACLR:MHITs<x>:BANDwidth? (Query Only) 1666
READ:LTE:ACLR:MHITs<x>:CHANnel:NAME? (Query Only) 1667
READ:LTE:ACLR:MHITs<x>:FREQuency:OFFSet? (Query Only) 1667
READ:LTE:ACLR:MHITs<x>:INTeg:LOWEr:ABSolute? (Query Only) 1668
READ:LTE:ACLR:MHITs<x>:INTeg:LOWEr:RELative? (Query Only) 1668
READ:LTE:ACLR:MHITs<x>:INTeg:UPPEr:ABSolute? (Query Only) 1669
READ:LTE:ACLR:REFerence:POWer? (Query Only) 1669
READ:LTE:ACLR:RESUlts:STATus? (Query Only) 1669
READ:LTE:ACLR:SPECtrum:X? (Query Only) 1670
READ:LTE:ACLR:SPECtrum:XY? (Query Only) 1670
READ:LTE:ACLR:SPECtrum:Y? (Query Only) 1671
READ:LTE:CHSPectrum:OBW? (Query Only) 1671
READ:LTE:CHSPectrum:POWer:CHANnel? (Query Only) 1672
READ:LTE:CHSPectrum:SPECtrum? (Query Only) 1672
READ:LTE:CONSte:CELL:ID? (Query Only) 1673
READ:LTE:CONSte:FREQuency:ERRor? (Query Only) 1673
READ:LTE:CONSte:GROUp:ID? (Query Only) 1673
READ:LTE:CONSte:SECTor:ID? (Query Only) 1674
READ:LTE:PVTime:OFFSlot:POWer? (Query Only) 1674
READ:LTE:PVTime:RESUlts:STATus 1675

READ:LTE:PVTime:TRACe:X? (Query Only) 1675
READ:LTE:PVTime:TRACe:XY? (Query Only) 1675
READ:LTE:PVTime:TRACe:Y? (Query Only) 1676
READ:MCPower:ADJacent:CHANnels? (Query Only) 1676
READ:MCPower:CHANnel:POWer? (Query Only) 1677
READ:MCPower:MAIN:CHANnels? (Query Only) 1678
READ:MCPower:SPECtrum? (Query Only) 1678
READ:MERRor[:MEASview<y>]:FERRor? (Query only) 1679
READ:MERRor[:MEASview<y>]:PEAK? (Query only) 1680
READ:MERRor[:MEASview<y>]:PINDex? (Query only) 1680
READ:MERRor[:MEASview<y>]:RMS? (Query only) 1681
READ:MERRor[:MEASview<y>]:TRACe? (Query only) 1681
READ:NOISe:TABLe:VALue? (Query Only) 1682
READ:OBWidth:BOBW:XDBBandwidth? (Query Only) 1682
READ:OBWidth:BOBW:XDBMbandwidth:IN:BANDwidth? (Query Only) 1683
READ:OBWidth:BOBW:XDBMbandwidth:IN:LEFT:FREQuency? (Query Only) 1684
READ:OBWidth:BOBW:XDBMbandwidth:IN:RIGHt:FREQuency? (Query Only) 1684
READ:OBWidth:BOBW:XDBMbandwidth:OUT:BANDwidth? (Query Only) 1685
READ:OBWidth:BOBW:XDBMbandwidth:OUT:LEFT:FREQuency? (Query Only) 1685
READ:OBWidth:BOBW:XDBMbandwidth:OUT:RIGHt:FREQuency? (Query Only) 1686
READ:OBWidth:FREQuency:ERRor? (Query Only) 1686
READ:OBWidth:OBWidth:BANDwidth? (Query Only) 1687
READ:OBWidth:OBWidth:LEFT:FREQuency? (Query Only) 1687
READ:OBWidth:OBWidth:LEFT:LEVel? (Query Only) 1688
READ:OBWidth:OBWidth:POWer? (Query Only) 1688
READ:OBWidth:OBWidth:RIGHt:FREQuency? (Query Only) 1689
READ:OBWidth:OBWidth:RIGHt:LEVel? (Query Only) 1689
READ:OBWidth:SPECtrum? (Query Only) 1690
READ:OBWidth:XDBBandwidth:BANDwidth? (Query Only) 1690
READ:OBWidth:XDBBandwidth:LEFT:FREQuency? (Query Only) 1691
READ:OBWidth:XDBBandwidth:LEFT:LEVel? (Query Only) 1691
READ:OBWidth:XDBBandwidth:POWer? (Query Only) 1692
READ:OBWidth:XDBBandwidth:RIGHt:FREQuency? (Query Only) 1692
READ:OBWidth:XDBBandwidth:RIGHt:LEVel? (Query Only) 1693
READ:OFDM:APOWer? (Query Only) 1693
READ:OFDM:APOWer:PEAK? (Query Only) 1694
READ:OFDM:CONSte:MAGNitude? (Query Only) 1694
READ:OFDM:CONSte:PHASe? (Query Only) 1695
READ:OFDM:CONSte:TYPE? (Query Only) 1696
READ:OFDM:CONSte:VALue? (Query Only) 1697
READ:OFDM:CPE? (Query Only) 1697
READ:OFDM:CRESponse:MAGNitude? (Query Only) 1698
READ:OFDM:CRESponse:PHASe? (Query Only) 1698
READ:OFDM:EVM:PEAK:DECibel:ALL? (Query Only) 1699
READ:OFDM:EVM:PEAK:DECibel:DATA? (Query Only) 1699
READ:OFDM:EVM:PEAK:DECibel:PILots? (Query Only) 1700

READ:OFDM:EVM:PEAK:PERCent:ALL? (Query Only) . . . 1700
READ:OFDM:EVM:PEAK:PERCent:DATA? (Query Only) . . . 1701
READ:OFDM:EVM:PEAK:PERCent:PILots? (Query Only) . . . 1701
READ:OFDM:EVM:PEAK:SCARrier:ALL? (Query Only) . . . 1702
READ:OFDM:EVM:PEAK:SCARrier:DATA? (Query Only) . . . 1702
READ:OFDM:EVM:PEAK:SCARrier:PILots? (Query Only) . . . 1702
READ:OFDM:EVM:PEAK:SYMBol:ALL? (Query Only) . . . 1703
READ:OFDM:EVM:PEAK:SYMBol:DATA? (Query Only) . . . 1703
READ:OFDM:EVM:PEAK:SYMBol:PILots? (Query Only) . . . 1704
READ:OFDM:EVM:RMS:DECibel:ALL? (Query Only) . . . 1704
READ:OFDM:EVM:RMS:DECibel:DATA? (Query Only) . . . 1705
READ:OFDM:EVM:RMS:DECibel:PILots? (Query Only) . . . 1705
READ:OFDM:EVM:RMS:PERCent:ALL? (Query Only) . . . 1706
READ:OFDM:EVM:RMS:PERCent:DATA? (Query Only) . . . 1706
READ:OFDM:EVM:RMS:PERCent:PILots? (Query Only) . . . 1706
READ:OFDM:EVM:TRACe<x>? (Query Only) . . . 1707
READ:OFDM:FERRor? (Query Only) . . . 1708
READ:OFDM:FLATness:PASS? (Query Only) . . . 1708
READ:OFDM:FLATness:RESult? (Query Only) . . . 1708
READ:OFDM:FLATness:TRACe<x>? (Query Only) . . . 1709
READ:OFDM:GAIN:IMBalance? (Query Only) . . . 1710
READ:OFDM:IQ:ORIGin:OFFSet? (Query Only) . . . 1710
READ:OFDM:MERRor:TRACe<x>? (Query Only) . . . 1711
READ:OFDM:PACKet:DIRection? (Query Only) . . . 1711
READ:OFDM:PERRor:TRACe<x>? (Query Only) . . . 1712
READ:OFDM:POWer:TRACe<x>? (Query Only) . . . 1712
READ:OFDM:QUADrature:OFFSet? (Query Only) . . . 1713
READ:OFDM:SCARriers? (Query Only) . . . 1714
READ:OFDM:SCARriers:SPACing? (Query Only) . . . 1714
READ:OFDM:STABle:VALUe? (Query Only) . . . 1714
READ:OFDM:SYMBol:CERRor? (Query Only) . . . 1715
READ:OFDM:SYMBol:COUNt? (Query Only) . . . 1715
READ:P25:CONSte:FERRor? (Query Only) . . . 1716
READ:P25:EDIagram:FDEViation? (Query Only) . . . 1716
READ:P25:EDIagram:FERRor? (Query Only) . . . 1717
READ:P25:PVTime:BURSt:POWer? (Query Only) . . . 1717
READ:P25:PVTime:FALL:TIMe? (Query Only) . . . 1718
READ:P25:PVTime:FTIMe? (Query Only) . . . 1718
READ:P25:PVTime:TRACe:X? (Query Only) . . . 1719
READ:P25:PVTime:TRACe:XY? (Query Only) . . . 1719
READ:P25:SUMMary:LIMIt:SRA (Query Only) . . . 1720
READ:P25:SUMMary:MODULation:MEASurement:FIDelity? (Query Only) . . . 1720
READ:P25:SUMMary:MODULation:MEASurement:FIDelity:STATus (Query Only) . . . 1721
READ:P25:SUMMary:MODULation:MEASurement:FREQuency: ACCUracy (No Query Form) . . . 1721
READ:P25:SUMMary:MODULation:MEASurement:FREQuency: ACCUracy? (Query Only) . . . 1721
READ:P25:SUMMary:MODULation:MEASurement: FREQuency: ACCUracy:STATus (Query Only) . . . 1722

READ:P25:SUMMary:MODULation:MEASurement:FREQuency: DEVIation:NEGPeak . . . 1722
READ:P25:SUMMary:MODULation:MEASurement:FREQuency: DEVIation:NEGPeak:STATus . . . 1723
READ:P25:SUMMary:MODULation:MEASurement:FREQuency: DEVIation:POSPeak . . . 1723
READ:P25:SUMMary:MODULation:MEASurement:FREQuency: DEVIation:POSPeak:STATus . . . 1724
READ:P25:SUMMary:MODULation:MEASurement:FREQuency:ERRor . . . 1724
READ:P25:SUMMary:MODULation:MEASurement:FREQuency:ERRor (Query Only) . . . 1725
READ:P25:SUMMary:MODULation:MEASurement:SRA . . . 1725
READ:P25:SUMMary:MODULation:MEASurement:SRA:STATus (Query Only) . . . 1725
READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXOFF (Query Only) . . . 1726
READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXOFF:STATus? (Query Only) . . . 1726
READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXOFF? (Query Only) . . . 1727
READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXON (Query Only) . . . 1727
READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXON:STATus (Query Only) . . . 1728
READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXON? (Query Only) . . . 1728
READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXSs (Query Only) . . . 1728
READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXSs:STATus (Query Only) . . . 1729
READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXSs? (Query Only) . . . 1729
READ:P25:SUMMary:POWEr:MEASurement:HCPM:MINImum (Query Only) . . . 1730
READ:P25:SUMMary:POWEr:MEASurement:HCPM:MINImum:STATus? (Query Only) . . . 1730
READ:P25:SUMMary:POWEr:MEASurement:HCPM:MINImum? (Query Only) . . . 1731
READ:P25:SUMMary:POWEr:MEASurement:HCPM:OFFSlot (Query Only) . . . 1731
READ:P25:SUMMary:POWEr:MEASurement:HCPM:OFFSlot:STATus? (Query Only) . . . 1731
READ:P25:SUMMary:POWEr:MEASurement:HCPM:PEAK:ACPR:HI (Query Only) . . . 1732
READ:P25:SUMMary:POWEr:MEASurement:HCPM:PEAK:ACPR:LOW (Query Only) . . . 1732
READ:P25:SUMMary:POWEr:MEASurement:HCPM:PEAK:ACPR:MIN (Query Only) . . . 1733
READ:P25:SUMMary:POWEr:MEASurement:HCPM:PEAK:ACPR: MIN:STATus(Query Only) . . . 1733
READ:P25:SUMMary:POWEr:MEASurement:RF:STATus (Query Only) . . . 1734
READ:P25:SUMMary:POWEr:MEASurement:RF (Query Only) . . . 1734
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME: ALIGNment:TERRor:ONE (Query Only) . . . 1734
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME: ALIGNment:TERRor:ONE:ACQ:COUNT? (Query Only) . . . 1735
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME: ALIGNment:TERRor:ONE:STATus (Query Only) . . . 1735
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME: ALIGNment:TERRor:ONE?(Query Only) . . . 1736
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME: ALIGNment:TERRor:ZERO (Query Only) . . . 1736
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME: ALIGNment:TERRor:ZERO:ACQ:COUNT? (Query Only) . . . 1737
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME: ALIGNment:TERRor:ZERO:STATus (Query Only) 1737
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME: ALIGNment:TERRor:ZERO? (Query Only) . . . 1737
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME: ALIGNment:TOBSync (Query Only) . . . 1738
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck: TIME:ACQ:COUNT (Query Only) . . . 1738
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME: ALIGNment:TOBSync? (Query Only) . . . 1739
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck: TIME:ACQ:COUNT (Query Only) . . . 1739
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck: TIME:ACQ:COUNT:BI (Query Only) . . . 1740
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME: ALIGNment:TOBSync:ACQ:COUNT (Query Only) 1740
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck: TIME:ENCOder:BI:STATus (Query Only) . . . 1740
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck: TIME:ACQ:COUNT:BI?(Query Only) . . . 1741
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck: TIME:ENCOder:STATus (Query Only) . . . 1741

READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck: TIME:ENCOder? (Query Only) 1742
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck: TIME:POWEr (Query Only) 1742
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1: ATTAck:TIME:POWEr:BI (Query Only) 1743
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck: TIME:POWEr:BI:STATus (Query Only) 1743
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck: TIME:POWEr:BI? (Query Only) 1743
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck: TIME:POWEr:STATus (Query Only) 1744
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck: TIME:POWEr? (Query Only) 1744
READ:P25:SUMMary:TRIGger:MEASurement: PHASe1:THRUput:DELAy (Query Only) 1745
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput: DELAy:ACQ:COUNT (Query Only) 1745
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1: THRUput:DELAy:ACQ:COUNT? (Query Only) 1746
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1: THRUput:DELAy:STATus (Query Only) 1746
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy? (Query Only) 1746
READ:PERRor[:MEASview<y>]:FERRor? (Query only) 1747
READ:PERRor[:MEASview<y>]:PEAK? (Query only) 1747
READ:PERRor[:MEASview<y>]:PINDex? (Query only) 1748
READ:PERRor[:MEASview<y>]:RMS (Query only) 1749
READ:PERRor[:MEASview<y>]:TRACe? (Query only) 1749
READ:PHVTime[:MEASview<y>]? (Query Only) 1750
READ:PHVTime[:MEASview<y>]:MAXimum? (Query Only) 1750
READ:PHVTime[:MEASview<y>]:MAXLocation? (Query Only) 1751
READ:PHVTime[:MEASview<y>]:MINimum? (Query Only) 1751
READ:PHVTime[:MEASview<y>]:MINLocation? (Query Only) 1752
READ:PHVTime[:MEASview<y>]:OGRAm? (Query Only) 1752
READ:PHVTime[:MEASview<y>]:RESult? (Query Only) 1753
READ:PHVTime[:MEASview<y>]:TRACe:BITMap? (Query Only) 1754
READ:PM:PNEGative? (Query Only) 1754
READ:PM:PPOSitive? (Query Only) 1755
READ:PM:PTPeak? (Query Only) 1755
READ:PM:RESult? (Query Only) 1756
READ:PM:RMS? (Query Only) 1756
READ:PNOise:ALL? (Query Only) 1757
READ:PNOise:CARRier:FERRor? (Query Only) 1757
READ:PNOise:CARRier:POWer? (Query Only) 1758
READ:PNOise:JITTer? (Query Only) 1758
READ:PNOise:RESidual:FM? (Query Only) 1759
READ:PNOise:RMS:PNOise? (Query Only) 1759
READ:PNOise:SPECtrum<x>:X? (Query Only) 1760
READ:PNOise:SPECtrum<x>[:Y]? (Query Only) 1760
READ:PNOise:SPECtrum<x>:XY? (Query Only) 1761
READ:PULSe:CUMulative:HISTogram[:MEASview<y>]:OUTLier:HIGHer? (Query Only) 1761
READ:PULSe:CUMulative:HISTogram[:MEASview<y>]:OUTLier:LOWer? (Query Only) 1762
READ:PULSe:CUMulative:HISTogram[:MEASview<y>]: PULse:COUNt? (Query Only) 1762
READ:PULSe:CUMulative:HISTogram[:MEASview<y>]:PULse:TOTal? (Query Only) 1763
READ:PULSe:CUMulative:STATistics[:MEASview<y>] (Query Only) 1763
READ:PULSe:OGRAm[:MEASview<y>]:MAX:FRAMe:COUNt (Query Only) 1765
READ:PULSe:OGRAm[:MEASview<y>]:RBW (Query Only) 1765

READ:PULSe[:RESult][:MEASview<y>]:ADVanced (Query Only) 1766
READ:PULSe[:RESult][:MEASview<y>]:ADVanced:RANGe (Query only) 1768
READ:PULSe:STATistics[:MEASview<y>]:ADVanced (Query Only) 1769
READ:PULSe[:RESult][:MEASview<y>]:ATX (Query Only) 1771
READ:PULSe[:RESult][:MEASview<y>]:AVERage (Query Only) 1772
READ:PULSe[:RESult][:MEASview<y>]:DRODb (Query Only) 1772
READ:PULSe[:RESult][:MEASview<y>]:DROop (Query Only) 1773
READ:PULSe[:RESult][:MEASview<y>]:DUTPct (Query Only) 1773
READ:PULSe[:RESult][:MEASview<y>]:DUTRatio (Query Only) 1774
READ:PULSe[:RESult][:MEASview<y>]:FABS (Query Only) 1774
READ:PULSe[:RESult][:MEASview<y>]:FALL (Query Only) 1775
READ:PULSe[:RESult][:MEASview<y>]:FDELta (Query Only) 1776
READ:PULSe[:RESult][:MEASview<y>]:FRDeviation (Query Only) 1776
READ:PULSe[:RESult][:MEASview<y>]:IRAMplitude (Query Only) 1777
READ:PULSe[:RESult][:MEASview<y>]:IRTime (Query Only) 1777
READ:PULSe[:RESult][:MEASview<y>]:MFReqerror (Query Only) 1778
READ:PULSe[:RESult][:MEASview<y>]:MPHerror (Query Only) 1778
READ:PULSe[:RESult][:MEASview<y>]:OVERDb (Query Only) 1779
READ:PULSe[:RESult][:MEASview<y>]:OVERshoot (Query Only) 1780
READ:PULSe[:RESult][:MEASview<y>]:PHDeviation (Query Only) 1780
READ:PULSe[:RESult][:MEASview<y>]:PPFD (Query Only) 1781
READ:PULSe[:RESult][:MEASview<y>]:PPFRequency (Query Only) 1781
READ:PULSe[:RESult][:MEASview<y>]:PPOWer (Query Only) 1782
READ:PULSe[:RESult][:MEASview<y>]:PPPD (Query Only) 1782
READ:PULSe[:RESult][:MEASview<y>]:PPPHase (Query Only) 1783
READ:PULSe[:RESult][:MEASview<y>]:RINTerval (Query Only) 1784
READ:PULSe[:RESult][:MEASview<y>]:RIPDb (Query Only) 1784
READ:PULSe[:RESult][:MEASview<y>]:RISE (Query Only) 1785
READ:PULSe[:RESult][:MEASview<y>]:RIPPle (Query Only) 1785
READ:PULSe[:RESult][:MEASview<y>]:RMSFreqerror (Query Only) 1786
READ:PULSe[:RESult][:MEASview<y>]:RMSPherror (Query Only) 1787
READ:PULSe[:RESult][:MEASview<y>]:RRATe (Query Only) 1787
READ:PULSe[:RESult][:MEASview<y>]:TIME (Query Only) 1788
READ:PULSe[:RESult][:MEASview<y>]:WIDTh (Query Only) 1788
READ:PULSe:STATistics[:MEASview<y>] (Query Only) 1789
READ:PULSe:STATistics[:MEASview<y>]:ADVanced (Query Only) 1790
READ:PULSe[:RESult]:ADVanced (Query Only) 1791
READ:PULSe:STATistics[:MEASview<y>]:AVERage (Query Only) 1793
READ:PULSe:STATistics[:MEASview<y>]:DRODb (Query Only) 1794
READ:PULSe:STATistics[:MEASview<y>]:DROop (Query Only) 1794
READ:PULSe:STATistics[:MEASview<y>]:DUTPct (Query Only) 1795
READ:PULSe:STATistics[:MEASview<y>]:DUTRatio (Query Only) 1796
READ:PULSe:STATistics[:MEASview<y>]:FABS (Query Only) 1796
READ:PULSe:STATistics[:MEASview<y>]:FALL (Query Only) 1797
READ:PULSe:STATistics[:MEASview<y>]:FDELta (Query Only) 1797
READ:PULSe:STATistics[:MEASview<y>]:FRDeviation (Query Only) 1798

READ:PULSe:STATistics[:MEASview<y>]:IRAMplitude (Query Only) 1799
READ:PULSe:STATistics[:MEASview<y>]:IRTime (Query Only) 1799
READ:PULSe:STATistics[:MEASview<y>]:MAX:PULSe:NUMber (Query Only) 1800
READ:PULSe:STATistics[:MEASview<y>]:MFReqerror (Query Only) 1800
READ:PULSe:STATistics[:MEASview<y>]:MIN:PULSe:NUMber? (Query Only) 1801
READ:PULSe:STATistics[:MEASview<y>]:MPHerror (Query Only) 1802
READ:PULSe:STATistics[:MEASview<y>]:OVEDb (Query Only) 1802
READ:PULSe:STATistics[:MEASview<y>]:OVERshoot (Query Only) 1803
READ:PULSe:STATistics[:MEASview<y>]:PHDeviation (Query Only) 1804
READ:PULSe:STATistics[:MEASview<y>]:PPFD (Query Only) 1804
READ:PULSe:STATistics[:MEASview<y>]:PPFRequency (Query Only) 1805
READ:PULSe:STATistics[:MEASview<y>]:PPOWer (Query Only) 1805
READ:PULSe:STATistics[:MEASview<y>]:PPPD (Query Only) 1806
READ:PULSe:STATistics[:MEASview<y>]:PPPHase (Query Only) 1807
READ:PULSe:STATistics[:MEASview<y>]:RINTerval (Query Only) 1807
READ:PULSe:STATistics[:MEASview<y>]:RIPDb (Query Only) 1808
READ:PULSe:STATistics[:MEASview<y>]:RIPPle (Query Only) 1809
READ:PULSe:STATistics[:MEASview<y>]:RISE (Query Only) 1809
READ:PULSe:STATistics[:MEASview<y>]:RMSFreqerror (Query Only) 1810
READ:PULSe:STATistics[:MEASview<y>]:RMSPherror (Query Only) 1811
READ:PULSe:STATistics[:MEASview<y>]:RRATe (Query Only) 1811
READ:PULSe:STATistics[:MEASview<y>]:WIDTh (Query Only) 1812
READ:PULSe:STATistics[:MEASview<y>]:X (Query Only) 1812
READ:PULSe:STATistics[:MEASview<y>]:XY (Query Only) 1813
READ:PULSe:TRACe[:MEASview<y>]:Y 1814
READ:PULSe:TRACe[:MEASview<y>]:X 1815
READ:PULSe:TRACe[:MEASview<y>]:XY 1815
READ:SEM:MHITs:COUNt? (Query Only) 1816
READ:SEM:MHITs<x>:AMPLitude:ABSolute? (Query Only) 1816
READ:SEM:MHITs<x>:AMPLitude:RELative? (Query Only) 1817
READ:SEM:MHITs<x>:FREQuency? (Query Only) 1817
READ:SEM:MHITs<x>:INTeg:ABSolute? (Query Only) 1818
READ:SEM:MHITs<x>:INTeg:RELative? (Query Only) 1818
READ:SEM:MHITs<x>:MARGin:ABSolute? (Query Only) 1819
READ:SEM:MHITs<x>:MARGin:RELative? (Query Only) 1819
READ:SEM:MHITs<x>:OFFSet? (Query Only) 1820
READ:SEM:PASS? (Query Only) 1820
READ:SEM:REF:POWer? (Query Only) 1821
READ:SEM:SPECtrum:X? (Query Only) 1821
READ:SEM:SPECtrum:XY? (Query Only) 1822
READ:SEM:SPECtrum[:Y]? (Query Only) 1822
READ:SGRam[:MEASview<y>] (Query Only) 1823
READ:SPECtrum[:MEASview<y>]:TRACe<x>? (Query Only) 1823
READ:SPECtrum[:MEASview<y>]:TRACe:BITMap? (Query Only) 1824
READ:SPURious:CARRier:POWer? (Query Only) 1825
READ:SPURious:COUNt? (Query Only) 1825

READ:SPURious:PASS? (Query Only) 1826
READ:SPURious:SPECtrum:X? (Query Only) 1826
READ:SPURious:SPECtrum:XY? (Query Only) 1827
READ:SPURious:SPECtrum[:Y]? (Query Only) 1827
READ:SPURious:SPUR<x>:AMPLitude:ABSolute? (Query Only) 1828
READ:SPURious:SPUR<x>:AMPLitude:RELative? (Query Only) 1828
READ:SPURious:SPUR<x>:FREQuency:ABSolute? (Query Only) 1829
READ:SPURious:SPUR<x>:FREQuency:RELative? (Query Only) 1829
READ:SPURious:SPUR<x>:LIMit:ABSolute? (Query Only) 1830
READ:SPURious:SPUR<x>:LIMit:RELative? (Query Only) 1830
READ:SPURious:SPUR<x>:LIMit:VIOLation? (Query Only) 1831
READ:SPURious:SPUR<x>:RANGe? (Query Only) 1831
READ:SQUality[:MEASview<y>]:FREQuency:DEViation? (Query only) 1832
READ:SQUality[:MEASview<y>]:FREQuency:DEViation:TABLe? (Query only) 1832
READ:SQUality[:MEASview<y>]:FREQuency:ERRor? (Query only) 1833
READ:SQUality[:MEASview<y>]:GAIN:IMBalance? (Query only) 1834
READ:SQUality[:MEASview<y>]:ORIGin:OFFSet? (Query only) 1834
READ:SQUality[:MEASview<y>]:PEAK:EVM? (Query only) 1835
READ:SQUality[:MEASview<y>]:PEAK:EVM:DB? (Query only) 1835
READ:SQUality[:MEASview<y>]:PEAK:EVM:DB:OFFSet? (Query only) 1836
READ:SQUality[:MEASview<y>]:PEAK:EVM:LOCation? (Query only) 1836
READ:SQUality[:MEASview<y>]:PEAK:EVM:LOCation:OFFSet? (Query only) 1837
READ:SQUality[:MEASview<y>]:PEAK:EVM:OFFSet? (Query only) 1838
READ:SQUality[:MEASview<y>]:PEAK:FERRor? (Query only) 1838
READ:SQUality[:MEASview<y>]:PEAK:MERRor? (Query only) 1839
READ:SQUality[:MEASview<y>]:PEAK:MERRor:DB? (Query only) 1839
READ:SQUality[:MEASview<y>]:PEAK:MERRor:LOCation? (Query only) 1840
READ:SQUality[:MEASview<y>]:PEAK:PERRor? (Query only) 1841
READ:SQUality[:MEASview<y>]:PEAK:PERRor:LOCation? (Query only) 1841
READ:SQUality[:MEASview<y>]:QUADrature:ERRor? (Query only) 1842
READ:SQUality[:MEASview<y>]:RHO? (Query only) 1842
READ:SQUality[:MEASview<y>]:RMS:EVM? (Query only) 1843
READ:SQUality[:MEASview<y>]:RMS:EVM:DB? (Query only) 1843
READ:SQUality[:MEASview<y>]:RMS:EVM:DB:OFFSet? (Query only) 1844
READ:SQUality[:MEASview<y>]:RMS:EVM:OFFSet? (Query only) 1844
READ:SQUality[:MEASview<y>]:RMS:FERRor? (Query only) 1845
READ:SQUality[:MEASview<y>]:RMS:MER:DB? (Query only) 1846
READ:SQUality[:MEASview<y>]:RMS:MERRor? (Query only) 1846
READ:SQUality[:MEASview<y>]:RMS:MERRor:DB? (Query only) 1847
READ:SQUality[:MEASview<y>]:RMS:PERRor? (Query only) 1847
READ:SQUality[:MEASview<y>]:SYMBol:LENGth? (Query only) 1848
READ:SQUality[:MEASview<y>]:SYMBol:RATE? (Query only) 1848
READ:SQUality[:MEASview<y>]:SYMBol:RATE:ERRor? (Query only) 1849
READ:TDIagram[:MEASview<y>]:FERRor? (Query only) 1849
READ:TDIagram[:MEASview<y>]:TRACe? (Query only) 1850
READ:TOVerview[:MEASview<y>] (Query Only) 1851

READ:WLAN:AD:CONSte:DATA:EVM? (Query Only) 1851
READ:WLAN:AD:CONSte:DATA:EVM:PEAK? (Query Only) 1852
READ:WLAN:AD:CONSte:DATA:EVM:PEAK:SYMBol? (Query Only) 1852
READ:WLAN:AD:CONSte:FERRor? (Query Only) 1853
READ:WLAN:AD:CONSte:IQDC:OFFSet? (Query Only) 1853
READ:WLAN:AD:EVM:TRACe? (Query Only) 1853
READ:WLAN:AD:SUMMary:AVERage:BURSt:COUNt (Query Only) 1854
READ:WLAN:AD:SUMMary:AVERage:BURSt:TOTAl:COUNt (Query Only) 1854
READ:WLAN:AD:SUMMary:ESNR (Query Only) 1855
READ:WLAN:AD:SUMMary:EVM:DATA:AVERage (Query Only) 1855
READ:WLAN:AD:SUMMary:EVM:DATA:AVERage:STATus (Query Only) 1856
READ:WLAN:AD:SUMMary:EVM:DATA:MAX (Query Only) 1856
READ:WLAN:AD:SUMMary:EVM:PACKet:CEF:EVM (Query Only) 1857
READ:WLAN:AD:SUMMary:EVM:PACKet:CEF:PEAK (Query Only) 1857
READ:WLAN:AD:SUMMary:EVM:PACKet:CEF:SYMBol (Query Only) 1858
READ:WLAN:AD:SUMMary:EVM:PACKet:DATA:EVM (Query Only) 1858
READ:WLAN:AD:SUMMary:EVM:PACKet:DATA:PEAK (Query Only) 1859
READ:WLAN:AD:SUMMary:EVM:PACKet:DATA:STATus (Query Only) 1859
READ:WLAN:AD:SUMMary:EVM:PACKet:DATA:SYMBol (Query Only) 1860
READ:WLAN:AD:SUMMary:EVM:PACKet:GUARd:EVM (Query Only) 1860
READ:WLAN:AD:SUMMary:EVM:PACKet:GUARd:PEAK (Query Only) 1861
READ:WLAN:AD:SUMMary:EVM:PACKet:GUARd:SYMBol (Query Only) 1861
READ:WLAN:AD:SUMMary:EVM:PACKet:HEADer:EVM (Query Only) 1861
READ:WLAN:AD:SUMMary:EVM:PACKet:HEADer:PEAK (Query Only) 1862
READ:WLAN:AD:SUMMary:EVM:PACKet:HEADer:SYMBol (Query Only) 1862
READ:WLAN:AD:SUMMary:EVM:PACKet:STF:EVM (Query Only) 1863
READ:WLAN:AD:SUMMary:EVM:PACKet:STF:PEAK (Query Only) 1863
READ:WLAN:AD:SUMMary:EVM:PACKet:STF:SYMBol (Query Only) 1864
READ:WLAN:AD:SUMMary:FERRor:AVERage (Query Only) 1864
READ:WLAN:AD:SUMMary:FERRor:AVERage:STATus (Query Only) 1865
READ:WLAN:AD:SUMMary:FERRor:HERTz (Query Only) 1865
READ:WLAN:AD:SUMMary:FERRor:MAX (Query Only) 1866
READ:WLAN:AD:SUMMary:FERRor:PPM (Query Only) 1866
READ:WLAN:AD:SUMMary:FERRor:STATus (Query Only) 1867
READ:WLAN:AD:SUMMary:IQ:GAIN:IMBALance (Query Only) 1867
READ:WLAN:AD:SUMMary:IQ:ORIGin:OFFSet (Query Only) 1868
READ:WLAN:AD:SUMMary:IQ:QUADrature:ERROr (Query Only) 1868
READ:WLAN:AD:SUMMary:PACKet:AGGRegation (Query Only) 1869
READ:WLAN:AD:SUMMary:PACKet:BTR (Query Only) 1869
READ:WLAN:AD:SUMMary:PACKet:HCS (Query Only) 1869
READ:WLAN:AD:SUMMary:PACKet:LENGth (Query Only) 1870
READ:WLAN:AD:SUMMary:PACKet:MCS (Query Only) 1870
READ:WLAN:AD:SUMMary:PACKet:PPDU (Query Only) 1871
READ:WLAN:AD:SUMMary:PACKet:RSSI (Query Only) 1871
READ:WLAN:AD:SUMMary:PACKet:SCRAmbler:INITialization (Query Only) 1872
READ:WLAN:AD:SUMMary:PACKet:TAROUND (Query Only) 1872

READ:WLAN:AD:SUMMary:PACKet:TLEN (Query Only) 1873
READ:WLAN:AD:SUMMary:PACKet:TYPE (Query Only) 1873
READ:WLAN:AD:SUMMary:RCPI (Query Only) 1873
READ:WLAN:AD:SUMMary:RF:POWer (Query Only) 1874
READ:WLAN:AY:CONSte:DATA:EVM? (Query Only) 1874
READ:WLAN:AY:CONSte:DATA:EVM:PEAK? (Query Only) 1875
READ:WLAN:AY:CONSte:DATA:EVM:PEAK:SYMBol? (Query Only) 1875
READ:WLAN:AY:CONSte:FERRor? (Query Only) 1876
READ:WLAN:AY:CONSte:FREQuency:ERROr? (Query Only) 1876
READ:WLAN:AY:CONSte:IQDC:OFFSet? (Query Only) 1876
READ:WLAN:AY:EVM:TRACe? (Query Only) 1877
READ:WLAN:AY:SUMMary:AVERage:BURSt:COUNt? (Query Only) 1877
READ:WLAN:AY:SUMMary:AVERage:BURSt:TOTAl:COUNt? (Query Only) 1878
READ:WLAN:AY:SUMMary:ESNR? (Query Only) 1878
READ:WLAN:AY:SUMMary:EVM:DATA:AVERage? (Query Only) 1878
READ:WLAN:AY:SUMMary:EVM:DATA:AVERage:STATus? (Query Only) 1879
READ:WLAN:AY:SUMMary:EVM:DATA:MAX? (Query Only) 1879
READ:WLAN:AY:SUMMary:EVM:DATA:SIGma? (Query Only) 1880
READ:WLAN:AY:SUMMary:EVM:PACKet:DATA:EVM? (Query Only) 1880
READ:WLAN:AY:SUMMary:EVM:PACKet:DATA:PEAK? (Query Only) 1881
READ:WLAN:AY:SUMMary:EVM:PACKet:DATA:STATus? (Query Only) 1881
READ:WLAN:AY:SUMMary:EVM:PACKet:DATA:SYMBol? (Query Only) 1881
READ:WLAN:AY:SUMMary:EVM:PACKet:EDMG:CEF:EVM? (Query Only) 1882
READ:WLAN:AY:SUMMary:EVM:PACKet:EDMG:CEF:PEAK? (Query Only) 1882
READ:WLAN:AY:SUMMary:EVM:PACKet:EDMG:CEF:SYMBol? (Query Only) 1883
READ:WLAN:AY:SUMMary:EVM:PACKet:EDMG:STF:EVM? (Query Only) 1883
READ:WLAN:AY:SUMMary:EVM:PACKet:EDMG:STF:PEAK? (Query Only) 1884
READ:WLAN:AY:SUMMary:EVM:PACKet:EDMG:STF:SYMBol? (Query Only) 1884
READ:WLAN:AY:SUMMary:EVM:PACKet:GUARd:EVM? (Query Only) 1885
READ:WLAN:AY:SUMMary:EVM:PACKet:GUARd:PEAK? (Query Only) 1885
READ:WLAN:AY:SUMMary:EVM:PACKet:GUARd:SYMBol? (Query Only) 1886
READ:WLAN:AY:SUMMary:EVM:PACKet:HEADer:EVM? (Query Only) 1886
READ:WLAN:AY:SUMMary:EVM:PACKet:HEADer:PEAK? (Query Only) 1886
READ:WLAN:AY:SUMMary:EVM:PACKet:HEADer:SYMBol? (Query Only) 1887
READ:WLAN:AY:SUMMary:EVM:PACKet:PRE1Edmg:A1Header:EVM? (Query Only) 1887
READ:WLAN:AY:SUMMary:EVM:PACKet:PRE1Edmg:CEF1:EVM? (Query Only) 1888
READ:WLAN:AY:SUMMary:EVM:PACKet:PRE1Edmg:GUARd1:EVM? (Query Only) 1888
READ:WLAN:AY:SUMMary:EVM:PACKet:PRE1Edmg:HEADer1:EVM? (Query Only) 1889
READ:WLAN:AY:SUMMary:EVM:PACKet:PRE1Edmg:STF1:EVM? (Query Only) 1889
READ:WLAN:AY:SUMMary:EVM:PACKet:PRE2Edmg:A2Header:EVM? (Query Only) 1890
READ:WLAN:AY:SUMMary:EVM:PACKet:PRE2Edmg:CEF2:EVM? (Query Only) 1890
READ:WLAN:AY:SUMMary:EVM:PACKet:PRE2Edmg:GUARd2:EVM? (Query Only) 1890
READ:WLAN:AY:SUMMary:EVM:PACKet:PRE2Edmg:HEADer2:EVM? (Query Only) 1891
READ:WLAN:AY:SUMMary:EVM:PACKet:PRE2Edmg:STF2:EVM? (Query Only) 1891
READ:WLAN:AY:SUMMary:FERRor:AVERage? (Query Only) 1892
READ:WLAN:AY:SUMMary:FERRor:AVERage:STATus? (Query Only) 1892

READ:WLAN:AY:SUMMary:FERRor:HERTz? (Query Only) 1893
READ:WLAN:AY:SUMMary:FERRor:MAX? (Query Only) 1893
READ:WLAN:AY:SUMMary:FERRor:PPM? (Query Only) 1893
READ:WLAN:AY:SUMMary:FERRor:STATus? (Query Only) 1894
READ:WLAN:AY:SUMMary:IQ:GAIN:IMBALance? (Query Only) 1894
READ:WLAN:AY:SUMMary:IQ:ORIGin:OFFSet? (Query Only) 1895
READ:WLAN:AY:SUMMary:IQ:QUADrature:ERROr? (Query Only) 1895
READ:WLAN:AY:SUMMary:PACKet:BEAMformed? (Query Only) 1895
READ:WLAN:AY:SUMMary:PACKet:BWBitmap? (Query Only) 1896
READ:WLAN:AY:SUMMary:PACKet:COMPbw? (Query Only) 1896
READ:WLAN:AY:SUMMary:PACKet:CRC? (Query Only) 1897
READ:WLAN:AY:SUMMary:PACKet:EDMGmcs? (Query Only) 1897
READ:WLAN:AY:SUMMary:PACKet:LDPCcwlen? (Query Only) 1897
READ:WLAN:AY:SUMMary:PACKet:PRIMarychannel? (Query Only) 1898
READ:WLAN:AY:SUMMary:PACKet:PSDUlen? (Query Only) 1898
READ:WLAN:AY:SUMMary:PACKet:RSSI? (Query Only) 1899
READ:WLAN:AY:SUMMary:PACKet:SCRAmbler:INITialization? (Query Only) 1899
READ:WLAN:AY:SUMMary:PACKet:SPATialstreams? (Query Only) 1899
READ:WLAN:AY:SUMMary:PACKet:STBCapplied? (Query Only) 1900
READ:WLAN:AY:SUMMary:PACKet:TRNLen? (Query Only) 1900
READ:WLAN:AY:SUMMary:RCPI? (Query Only) 1900
READ:WLAN:AY:SUMMary:RF:POWer? (Query Only) 1901
READ:WLAN:AY:SUMMary:SIG:QUAlity? (Query Only) 1901
READ:WLAN:CONSte:MAGNitude? (Query Only) 1902
READ:WLAN:CONSte:PHASe? (Query Only) 1902
READ:WLAN:CONSte:TYPE? (Query Only) 1903
READ:WLAN:CONSte:VALue? (Query Only) 1903
READ:WLAN:CRESponse:MAGNitude? (Query Only) 1904
READ:WLAN:CRESponse:PHASe? (Query Only) 1904
READ:WLAN:EVM:TRACe<x>? (Query Only) 1905
READ:WLAN:FLATness:PASS? (Query Only) 1906
READ:WLAN:FLATness:RESult? (Query Only) 1906
READ:WLAN:FLATness:TRACe<x>? (Query Only) 1907
READ:WLAN:MERRor:TRACe<x>? (Query Only) 1907
READ:WLAN:PERRor:TRACe<x>? (Query Only) 1908
READ:WLAN:PVTime:BURSt:POWer? (Query Only) 1908
READ:WLAN:PVTime:BURSt:WIDTh? (Query Only) 1909
READ:WLAN:PVTime:FALL:TIME? (Query Only) 1909
READ:WLAN:PVTime:FALL:TIME:STARt? (Query Only) 1910
READ:WLAN:PVTime:FALL:TIME:STOP? (Query Only) 1910
READ:WLAN:PVTime:FTIMe? (Query Only) 1910
READ:WLAN:PVTime:RISE:TIME? (Query Only) 1911
READ:WLAN:PVTime:RISE:TIME:STARt? (Query Only) 1911
READ:WLAN:PVTime:RISE:TIME:STOP? (Query Only) 1912
READ:WLAN:PVTime:RTIMe? (Query Only) 1912
READ:WLAN:PVTime:TRACe:X? (Query Only) 1912

READ:WLAN:PVTime:TRACe:XY? (Query Only) 1913
READ:WLAN:PVTime:TRACe[:Y]? (Query Only) 1914
READ:WLAN:STABle:VALUe? (Query Only) 1914
READ:WLAN:SUMMary:BURSt:INDex? (Query Only) 1915
READ:WLAN:SUMMary:BURSt:POWer? (Query Only) 1915
READ:WLAN:SUMMary:BURSt:POWer:CFACtor? (Query Only) 1915
READ:WLAN:SUMMary:CERRor? (Query Only) 1916
READ:WLAN:SUMMary:CPE? (Query Only) 1916
READ:WLAN:SUMMary:EVM:PEAK:ALL? (Query Only) 1917
READ:WLAN:SUMMary:EVM:PEAK:CHIP:ALL? (Query Only) 1917
READ:WLAN:SUMMary:EVM:PEAK:CHIP:ONEKchips? (Query Only) 1917
READ:WLAN:SUMMary:EVM:PEAK:DATA? (Query Only) 1918
READ:WLAN:SUMMary:EVM:PEAK:ONEKchips? (Query Only) 1918
READ:WLAN:SUMMary:EVM:PEAK:PILots? (Query Only) 1919
READ:WLAN:SUMMary:EVM:PEAK:SCARrier:ALL? (Query Only) 1919
READ:WLAN:SUMMary:EVM:PEAK:SCARrier:DATA? (Query Only) 1920
READ:WLAN:SUMMary:EVM:PEAK:SCARrier:PILots? (Query Only) 1920
READ:WLAN:SUMMary:EVM:PEAK:SYMBol:ALL? (Query Only) 1920
READ:WLAN:SUMMary:EVM:PEAK:SYMBol:DATA? (Query Only) 1921
READ:WLAN:SUMMary:EVM:PEAK:SYMBol:PILots? (Query Only) 1921
READ:WLAN:SUMMary:EVM:RMS:ALL? (Query Only) 1922
READ:WLAN:SUMMary:EVM:RMS:AVERage? (Query Only) 1922
READ:WLAN:SUMMary:EVM:RMS:DATA? (Query Only) 1922
READ:WLAN:SUMMary:EVM:RMS:MAXimum? (Query Only) 1923
READ:WLAN:SUMMary:EVM:RMS:ONEKchips? (Query Only) 1923
READ:WLAN:SUMMary:EVM:RMS:PILots? (Query Only) 1924
READ:WLAN:SUMMary:FERRor? (Query Only) 1924
READ:WLAN:SUMMary:HEADer? (Query Only) 1925
READ:WLAN:SUMMary:HTSig? (Query Only) 1925
READ:WLAN:SUMMary:IQOFfset? (Query Only) 1925
READ:WLAN:SUMMary:LSIG? (Query Only) 1926
READ:WLAN:SUMMary:PACKet? (Query Only) 1926
READ:WLAN:SUMMary:PACKet:DMODulation? (Query Only) 1926
READ:WLAN:SUMMary:PACKet:FORMat? (Query Only) 1927
READ:WLAN:SUMMary:PACKet:GUARd? (Query Only) 1927
READ:WLAN:SUMMary:SIG? (Query Only) 1928
READ:WLAN:SUMMary:VHTSig:A? (Query Only) 1928
READ:WLAN:SUMMary:VHTSig:B? (Query Only) 1929
Sense commands 1929
[SENSe]:ACPower:AVERage 1929
[SENSe]:ACPower:AVERage:COUNt 1930
[SENSe]:ACPower:{BANDwidth|BWIDth}[:RESolution] 1930
[SENSe]:ACPower:{BANDwidth|BWIDth}[:RESolution]:ACTual? (Query Only) 1930
[SENSe]:ACPower:{BANDwidth|BWIDth}[:RESolution]:AUTO 1931
[SENSe]:ACPower:{BANDwidth|BWIDth}:VIDeo 1931
[SENSe]:ACPower:{BANDwidth|BWIDth}:VIDeo:STATe 1932

[SENSe]:ACPower:CHANnel:{BANDwidth|BWIDth} 1932
[SENSe]:ACPower:CHANnel:FILTer 1933
[SENSe]:ACPower:CHANnel:PAIRs 1933
[SENSe]:ACPower:CHANnel:SPACing 1934
[SENSe]:ACPower:CHIPrate 1934
[SENSe]:ACPower:CLEar:RESults (No Query Form) 1934
[SENSe]:ACPower:FREQuency 1935
[SENSe]:ACPower:FREQuency:STEP 1935
[SENSe]:ACPower:FREQuency:STEP:AUTO 1936
[SENSe]:ACPower:NFLoor:STATe 1936
[SENSe]:ACPower:POINts:COUNt 1937
[SENSe]:ACPower:RRCRolloff 1937
[SENSe]:ACPower:SYMBrate 1938
[SENSe]:ACQuisition:{BANDwidth|BWIDth} 1938
[SENSe]:ACQuisition:FFRame:ACTual? (Query Only) 1939
[SENSe]:ACQuisition:FFRame:LIMit 1939
[SENSe]:ACQuisition:FFRame:STATe 1939
[SENSe]:ACQuisition:FSAVe:ENABle 1940
[SENSe]:ACQuisition:FSAVe:FILEs:MAXimum 1940
[SENSe]:ACQuisition:FSAVe:FORMat 1941
[SENSe]:ACQuisition:FSAVe:LOCation 1941
[SENSe]:ACQuisition:FSAVe:NAME:BASE 1942
[SENSe]:ACQuisition:MEMory:AVAilable:SAMPles? (Query Only) 1942
[SENSe]:ACQuisition:MEMory:CAPacity[:TIME]? (Query Only) 1942
[SENSe]:ACQuisition:MEMory:USED[:PERCent]? (Query Only) 1943
[SENSe]:ACQuisition:MODE 1943
[SENSe]:ACQuisition:RECord:ACQuisition:{BANDwidth|BWIDth}:AUTO 1944
[SENSe]:ACQuisition:RECord:ACQuisition:{BANDwidth|BWIDth} <value> 1944
[SENSe]:ACQuisition:RECord:FILE:LENGth 1945
[SENSe]:ACQuisition:RECord:FILE:LENGth:ENABle 1945
[SENSe]:ACQuisition:RECord:FILE:LENGth:UNITs 1946
[SENSe]:ACQuisition:RECord:FILE:LOCation 1946
[SENSe]:ACQuisition:RECord:FILE:MAXimum 1947
[SENSe]:ACQuisition:RECord:FILE:NAME:BASE 1947
[SENSe]:ACQuisition:RECord:FILE:SIZE? (Query Only) 1948
[SENSe]:ACQuisition:RECord:FILE:STRucture 1948
[SENSe]:ACQuisition:RECord:RESults? (Query Only) 1948
[SENSe]:ACQuisition:RECord:STARt (No Query Form) 1949
[SENSe]:ACQuisition:RECord:STOP (No Query Form) 1950
[SENSe]:ACQuisition:RECord:TYPE 1950
[SENSe]:ACQuisition:SAMPles 1950
[SENSe]:ACQuisition:SEConds 1951
[SENSe]:{AM|FM|PM}:{BANDwidth|BWIDth}:MEASurement 1951
[SENSe]:{AM|FM|PM}:CLEar:RESults (No Query Form) 1952
[SENSe]:{AM|FM|PM}:{MTPoints|MAXTracepoints} 1952
[SENSe]:AM:DETect:AMPLitude 1953

[SENSe]:ANALysis:ADVanced:DITHer . . . 1953
[SENSe]:ANALysis:ADVanced:DITHer:HWARe:STATus? (Query Only) . . . 1954
[:SENSe]:ANALysis:BANDwidth:FILTer:MODE {set and query} . . . 1954
[SENSe]:ANALysis:LENGth [:SOURce<z>] . . . 1955
[SENSe]:ANALysis:LENGth[:SOURce<z>]:ACTual? (Query only) . . . 1955
[SENSe]:ANALysis:LENGth[:SOURce<z>]:AUTO . . . 1956
[SENSe]:ANALysis:REFerence[:SOURce<z>] . . . 1956
[SENSe]:ANALysis:STARt[:SOURce<z>] . . . 1957
[SENSe]:ANALysis:STARt[:SOURce<z>]:AUTO . . . 1957
[SENSe]:ANSPectrum:STARt:AUTO:METHod . . . 1958
[SENSe]:AUDio:AVERage:COUNt . . . 1958
[SENSe]:AUDio:AVERage:ENABle . . . 1959
[SENSe]:AUDio:BANDwidth|BWIDth . . . 1959
[SENSe]:AUDio:BANDwidth|BWIDth:RESolution . . . 1960
[SENSe]:AUDio:BANDwidth|BWIDth:RESolution:AUTO . . . 1960
[SENSe]:AUDio:BANDwidth|BWIDth:RESolution:SHAPe . . . 1960
[SENSe]:AUDio:CARRier:OFFSet . . . 1961
[SENSe]:AUDio:CARRier:OFFSet:AUTO . . . 1961
[SENSe]:AUDio:DEMod:MUTE . . . 1962
[SENSe]:AUDio:DEMod[:STATe] . . . 1962
[SENSe]:AUDio:DEMod:STReam:ACTivate . . . 1963
[SENSe]:AUDio:DEMod:STReam[:FILE] (No Query Form) . . . 1963
[SENSe]:AUDio:DEMod:TYPE . . . 1964
[SENSe]:AUDio:DEMod:VOLume . . . 1964
[SENSe]:AUDio:FILTer:DEEMphasis . . . 1965
[SENSe]:AUDio:FILTer:DEEMphasis:ENABle . . . 1965
[SENSe]:AUDio:FILTer:DEEMphasis:USER . . . 1966
[SENSe]:AUDio:FILTer:HPF . . . 1966
[SENSe]:AUDio:FILTer:HPF:ENABle . . . 1967
[SENSe]:AUDio:FILTer:HPF:USER . . . 1967
[SENSe]:AUDio:FILTer:LPF . . . 1967
[SENSe]:AUDio:FILTer:LPF:ENABle . . . 1968
[SENSe]:AUDio:FILTer:LPF:USER . . . 1968
[SENSe]:AUDio:FILTer:MODE:PREDefined . . . 1969
[SENSe]:AUDio:FILTer:SOURce? (Query Only) . . . 1969
[SENSe]:AUDio:FILTer:STANdard . . . 1970
[SENSe]:AUDio:FILTer:STANdard:ENABle . . . 1970
[SENSe]:AUDio:FREQuency . . . 1971
[SENSe]:AUDio:FREQuency:AUTO . . . 1971
[SENSe]:AUDio:HNOise:ENABle . . . 1971
[SENSe]:AUDio:HNOise:REFerence (No Query Form) . . . 1972
[SENSe]:AUDio:SIGnal:TYPE . . . 1972
[SENSe]:AUDio:SPECtrum:CLEar:RESults (No Query Form) . . . 1973
[SENSe]:AUDio:SPECtrum:POINts:COUNt . . . 1973
[SENSe]:AUDio:UNITs . . . 1974
[SENSe]:AVTime[:MEASview<y>]:SELEcted:SOURce . . . 1975

[SENSe]:AVTime[:MEASview<y>]:SPAN:LINK 1977
[SENSe]:AVTime[:MEASview<y>]:{BANDwidth|BWIDth} 1978
[SENSe]:AVTime[:MEASview<y>]:{BANDwidth|BWIDth}:ACTual? (Query Only) 1978
[SENSe]:AVTime[:MEASview<y>]:CLEar:RESults (No Query Form) 1979
[SENSe]:AVTime[:MEASview<y>]:COLor 1979
[SENSe]:AVTime[:MEASview<y>]:COLor:MAXimum 1980
[SENSe]:AVTime[:MEASview<y>]:COLor:MINimum 1981
[SENSe]:AVTime[:MEASview<y>]:COLor:REVerse[:STATe] 1981
[SENSe]:AVTime[:MEASview<y>]:MAXTracepoints 1982
[SENSe]:AVTime[:MEASview<y>]:METHod 1982
[SENSe]:AVTime[:MEASview<y>]:OGRAm:COLor 1983
[SENSe]:AVTime[:MEASview<y>]:OGRAm:COLor:MAXimum 1984
[SENSe]:AVTime[:MEASview<y>]:OGRAm:COLor:MINimum 1984
[SENSe]:AVTime[:MEASview<y>]:OGRAm:COLor:REVerse 1985
[SENSe]:AVTime[:MEASview<y>]:PLOT 1986
[SENSe]:AVTime[:MEASview<y>]:SPAN 1986
[SENSe]:BLUEtooth:ANALysis:LENGth 1987
[SENSe]:BLUEtooth:ANALysis:LENGth:ACTual? (Query Only) 1987
[SENSe]:BLUEtooth:ANALysis:LENGth:AUTO 1988
[SENSe]:BLUEtooth:ANALysis:MEAS[:BANDwidth|BWIDth] 1988
[SENSe]:BLUEtooth:ANALysis:MEAS:SOURce 1989
[SENSe]:BLUEtooth:ANALysis:OFFSet 1989
[SENSe]:BLUEtooth:ANALysis:OFFSet:AUTO 1990
[SENSe]:BLUEtooth:ANALysis:TIMe:UNITs (No Query Form) 1990
[SENSe]:BLUEtooth([:BANDwidth]|:BWIDth):MODE 1990
[SENSe]:BLUEtooth([:BANDwidth]|:BWIDth):TINTerval 1991
[SENSe]:BLUEtooth([:BANDwidth]|:BWIDth):TINTerval:AUTO 1991
[SENSe]:BLUEtooth:CONStellation:PREFs:GRATicule:SHOW 1992
[SENSe]:BLUEtooth:CONStellation:PREFs:MARKERS:SHOW 1992
[SENSe]:BLUEtooth:CONStellation:PREFs:RADIx 1993
[SENSe]:BLUEtooth:CONStellation:TRACE<x>:CONTent 1993
[SENSe]:BLUEtooth:CONStellation:TRACe<x>:FREEze 1994
[SENSe]:BLUEtooth:CONSTellation:TRACE<x>:POINtsPerSymbol 1994
[SENSe]:BLUEtooth:CONStellation:TRACe<x>:SHOW 1995
[SENSe]:BLUEtooth:DELTa:AVERage:FONE 1995
[SENSe]:BLUEtooth:DELTa:AVERage:FTWO 1995
[SENSe]:BLUEtooth:FDVTime:TRACE:OCTEt? (Query Only) 1996
[SENSe]:BLUEtooth:FDVTime:TRACE:OCTEt:TOTAL:COUNt? (Query Only) 1996
[SENSe]:BLUEtooth:FDVTime:TRACE:VIEW:MODE 1997
[SENSe]:BLUEtooth:FILTer:ALPHa 1997
[SENSe]:BLUEtooth:FILTer:MEASurement 1998
[SENSe]:BLUEtooth:FILTer:REFerence 1998
[SENSe]:BLUEtooth:FREQuency:DEViation 1999
[SENSe]:BLUEtooth:FREQuency:DEViation:AUTO 1999
[SENSe]:BLUEtooth:FREQuency:ERRor 2000
[SENSe]:BLUEtooth:FREQuency:ERRor:AUTO 2000

[SENSe]:BLUEtooth:INBEmissions:POWEr:LIMIt<x> 2000
[SENSe]:BLUEtooth:POWer:CLASs 2001
[SENSe]:BLUEtooth:STANdard 2001
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:DELTa:FONE:AVERage:MAX 2002
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:DELTa:FONE:AVERage:MIN 2002
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:DELTa:FTWO:PERcent 2003
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:DELTa:FTWO:RESult 2003
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:MAX:DRIFt:FNFZero 2003
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:MAX:FREQuency:OFFSet 2004
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:MAX:FREQuency: OFFSet:PREAmBDR 2004
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs: ONE:AVERage:MAX 2005
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs: ONE:AVERage:MIN 2005
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs: ONE:PEAK:MAX 2006
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs: ONE:PEAK:MIN 2006
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs: THREe:AVERage:MAX 2007
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs: THREe:AVERage:MIN 2007
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs: THREe:PEAK:MAX 2008
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs: THREe:PEAK:MIN 2008
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs: TWO:AVERage:MAX 2008
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs: TWO:AVERage:MIN 2009
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs: TWO:PEAK:MAX 2009
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs: TWO:PEAK:MIN 2010
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:DELTa:FONE:AVERage:MIN 2010
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:DELTa:FTWO:PERcent 2011
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:DELTa:FTWO:RESult 2011
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:MAX:DRIFt:FNFZero 2011
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:MAX:FREQuency:OFFSet 2012
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:MAX:FREQuency: OFFSet:PREAmble 2012
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:OPOWer:AVERage:MAX 2013
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:OPOWer:AVERage:MIN 2013
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:OPOWer:PEAK:MAX 2014
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:DELTa:FONE:AVERage:MAX 2014
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:OPOWer:PEAK:MIN 2014
[SENSe]:BLUEtooth:SUMMary:LIMIt:MAX:DRIFt:F1FZero 2015
[SENSe]:BLUEtooth:SUMMary:LIMIt:MAX:DRIFt:FNFN5 2015
[SENSe]:BLUEtooth:SUMMary:LIMIt:MAX:DRIFt:LECoded:F0FThree 2016
[SENSe]:BLUEtooth:SUMMary:LIMIt:MAX:DRIFt:LECoded:F1FFour 2016
[SENSe]:BLUEtooth:SUMMary:LIMIt:MAX:DRIFt:LECoded:FNFN3 2017
[SENSe]:BLUEtooth:SYMBol:POINts 2017
[SENSe]:BLUEtooth:TEST:PATTern 2018
[SENSe]:BLUEtooth:TEST:PATTern:AUTO:DETEct 2018
[SENSe]:BLUEtooth:TIME:ZERO:REFERENCE 2019
[SENSe]:BLUEtooth:TRACe:TYPE 2019
[SENSe]:CONSte[:MEASview<y>]:SELEcted:SOURce 2020
[SENSe]:CCDF:{BANDwidth|BWIDth} 2020
[SENSe]:CCDF:CLEar (No Query Form) 2020

[SENSe]:CCDF:TIME:TOTal:LENGth 2021
[SENSe]:CCDF:TIME:TYPE 2021
[SENSe]:DDEMod[:SOURce<x>]:ANALysis:LENGth 2022
[SENSe]:DDEMod[:SOURce<x>]:ANALysis:LENGth:ACTual? (Query only) 2023
[SENSe]:DDEMod[:SOURce<x>]:ANALysis:LENGth:AUTO 2023
[SENSe]:DDEMod[:SOURce<x>]:{BANDwidth|BWIDth}:MODE 2024
[SENSe]:DDEMod[:SOURce<x>]:{BANDwidth|BWIDth}:TINTerval 2024
[SENSe]:DDEMod[:SOURce<x>]:{BANDwidth|BWIDth}:TINTerval:AUTO 2025
[SENSe]:DDEMod[:SOURce<x>]:BURSt:DETect 2026
[SENSe]:DDEMod[:SOURce<x>]:BURSt:THReshold 2026
[SENSe]:DDEMod[:SOURce<x>]:CARRier:OFFSet 2027
[SENSe]:DDEMod[:SOURce<x>]:CARRier:OFFSet:AUTO 2028
[SENSe]:DDEMod[:SOURce<x>]:EQUalizer:AVAilable? (Query only) 2028
[SENSe]:DDEMod[:SOURce<x>]:EQUalizer:CONVergence 2029
[SENSe]:DDEMod[:SOURce<x>]:EQUalizer:ENABle 2029
[SENSe]:DDEMod[:SOURce<x>]:EQUalizer:LENGth 2030
[SENSe]:DDEMod[:SOURce<x>]:EQUalizer:MODE 2030
[SENSe]:DDEMod[:SOURce<x>]:EQUalizer:RESet (No query form) 2031
[SENSe]:DDEMod[:SOURce<x>]:EQUalizer:TAPS 2031
[SENSe]:DDEMod[:SOURce<x>]:EQUalizer:TSRAtio 2031
[SENSe]:DDEMod[:SOURce<x>]:FILTer:ALPHa 2032
[SENSe]:DDEMod[:SOURce<x>]:FILTer:MEASurement 2032
[SENSe]:DDEMod:SOURCE<x>:FILTer:REFerence 2033
[SENSe]:DDEMod[:SOURce<x>]:FREQuency:DEViation 2034
[SENSe]:DDEMod[:SOURce<x>]:FREQuency:DEViation:AUTO 2035
[SENSe]:DDEMod[:SOURce<x>]:MAGNitude:NORMalize 2035
[SENSe]:DDEMod[:SOURce<x>]:MINDex 2036
[SENSe]:DDEMod[:SOURce<x>]:MINDex:AUTO 2037
[SENSe]:DDEMod[:SOURce<x>]:MODulation:TYPE 2037
[SENSe]:DDEMod[:SOURce<x>]:PRESet (No query form) 2039
[SENSe]:DDEMod[:SOURce<x>]:RING:RADius 2042
[SENSe]:DDEMod[:SOURce<x>]:RING:ROTation 2042
[SENSe]:DDEMod[:SOURce<x>]:SRATe 2043
[SENSe]:DDEMod[:SOURce<x>]:SWAP:IQ 2043
[SENSe]:DDEMod[:SOURce<x>]:SYMBol:HSSHift 2044
[SENSe]:DDEMod[:SOURce<x>]:SYMBol:MAP:SOURce? (Query only) 2044
[SENSe]:DDEMod[:SOURce<x>]:SYMBol:MAP[:STATe] 2045
[SENSe]:DDEMod[:SOURce<x>]:SYMBol:PLOT:POSition 2045
[SENSe]:DDEMod[:SOURce<x>]:SYMBol:POINts 2046
[SENSe]:DDEMod[:SOURce<x>]:SYMBol:RATE:SEARch 2047
[SENSe]:DDEMod[:SOURce<x>]:SYNCh:WORD 2047
[SENSe]:DDEMod[:SOURce<x>]:SYNCh:WORD:SYMBol 2048
[SENSe]:DDEMod[:SOURce<x>]:TIMe:UNITs 2048
[SENSe]:DIQVtime[:MEASview<y>]:SELEcted:SOURce 2049
[SENSe]:DPX:AUDio:DEMod:GAIN 2049
[SENSe]:DPX:AUDio:DEMod:RXBWidth 2050

[SENSe]:DPX:AUDio:DEMod:RXFRequency? (Query Only) 2050
[SENSe]:DPX:AUDio:DEMod:STATe 2051
[SENSe]:DPX:AUDio:DEMod:TUNE 2051
[SENSe]:DPX:AUDio:DEMod:TYPE 2052
[SENSe]:DPX:{BANDwidth|BWIDth}:ACTual? (Query Only) 2052
[SENSe]:DPX:{BANDwidth|BWIDth}:RESolution 2053
[SENSe]:DPX:{BANDwidth|BWIDth}[:RESolution]:AUTO 2053
[SENSe]:DPX:CLEar:RESults (No Query Form) 2053
[SENSe]:DPX:COLor 2054
[SENSe]:DPX:COLor:MAXimum 2055
[SENSe]:DPX:COLor:MINimum 2055
[SENSe]:DPX:COLor:REVerse[:STATe] 2056
[SENSe]:DPX:DDENsity:MEASurement:SHOW 2056
SENSe:DPX:DGRam:COLor 2056
[SENSe]:DPX:DGRam:COLor:MAXimum 2057
[SENSe]:DPX:DGRam:COLor:MINimum 2058
[SENSe]:DPX:DGRam:COLor:REVerse 2058
[SENSe]:DPX:DGRam:POINts:COUNt 2059
[SENSe]:DPX:DGRam:TIME:CAPacity? (Query Only) 2059
[SENSe]:DPX:DGRam:TIME:RESolution 2060
[SENSe]:DPX:DGRam:TIME:RESolution:AUTO 2060
[SENSe]:DPX:DGRam:WATerfall:DIRection 2061
[SENSe]:DPX:DGRam:WATerfall:ENABle 2061
[SENSe]:DPX:DGRam:Y[:SCALe]:AUTO (No Query Form) 2061
[SENSe]:DPX:DGRam:Y[:SCALe]:RESet (No Query Form) 2062
[SENSe]:DPX:DWELl 2062
[SENSe]:DPX:DWELl:AUTO 2063
[SENSe]:DPX:FREQuency:CENTer 2063
[SENSe]:DPX:FREQuency:SPAN 2064
[SENSe]:DPX:FREQuency:SPAN:{BANDwidth|BWIDth}[:RESolution]:RATio 2064
[SENSe]:DPX:FREQuency:STARt 2065
[SENSe]:DPX:FREQuency:STEP 2065
[SENSe]:DPX:FREQuency:STEP:AUTO 2066
[SENSe]:DPX:FREQuency:STOP 2066
[SENSe]:DPX:PLOT 2067
[SENSe]:DPX:POINts:COUNt 2067
[SENSe]:DPX:TDM:{BANDwidth|BWIDth} 2068
[SENSe]:DPX:TDM:{BANDwidth|BWIDth}:ACTual? (Query Only) 2068
[SENSe]:DPX:TDM:FREQuency:CENTer 2069
[SENSe]:DPX:TDM:FREQuency:TXBer:CLEar:RESults (No Query Form) 2069
[SENSe]:DPX:TDM:FREQuency:TXBer:ENABle 2070
[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern 2070
[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern:INVert 2071
[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern:REPeat 2071
[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern:REVerse 2072
[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern:SOURce? (Query Only) 2072

[SENSe]:DPX:TDM:RBW . . . 2073
[SENSe]:DPX:TDM:RBW:STATe . . . 2073
[SENSe]:DPX:TDM:SWEep:TIME . . . 2074
[SENSe]:DPX:TDM:TMOTion . . . 2074
[SENSe]:EDIagram[:MEASview<y>]:SELEcted:SOURce . . . 2074
[SENSe]:DEMOd:EVM[:MEASview<y>]:SELEcted:SOURce . . . 2075
SENSe:EMC:ACCEssory<x>:ANTEnna:DISTance . . . 2075
SENSe:EMC:ACCEssory<x>:ASSEt:TAG . . . 2076
SENSe:EMC:ACCEssory<x>:CALibration:DATE . . . 2076
SENSe:EMC:ACCEssory<x>:CALibration:DATE:DISAble . . . 2077
SENSe:EMC:ACCEssory<x>:CONStant:LOSS . . . 2077
SENSe:EMC:ACCEssory<x>:ENABle . . . 2078
SENSe:EMC:ACCEssory<x>:FREQuency:INTerpolation . . . 2078
SENSe:EMC:ACCEssory<x>:LOSS:TABLe:ADD (No Query Form) . . . 2079
SENSe:EMC:ACCEssory<x>:LOSS:TABLe:DELete:ALL (No Query Form) . . . 2080
SENSe:EMC:ACCEssory<x>:LOSS:TYPE . . . 2080
SENSe:EMC:ACCEssory<x>:MODel . . . 2081
SENSe:EMC:ACCEssory<x>:SERial . . . 2081
SENSe:EMC:ACCEssory<x>:TYPE . . . 2082
SENSe:EMC:CLEar:RESults (No Query Form) . . . 2082
SENSe:EMC:EMISsion:TYPE . . . 2083
SENSe:EMC[:FREQuency]:OVERlap? (Query Only) . . . 2083
SENSe:EMC:LIST . . . 2084
SENSe:EMC:MARGin . . . 2084
SENSe:EMC:MEASurement:AMBIent:DELTa:THREshold . . . 2085
SENSe:EMC:MEASurement:AMBIent:DETEctor . . . 2085
SENSe:EMC:MEASurement:AMBIent:DWELl:TIME . . . 2086
SENSe:EMC:MEASurement:AMBIent:DWELl:TIME:AUTO . . . 2086
SENSe:EMC:MEASurement:AMBIent:STARt (No Query Form) . . . 2087
SENSe:EMC:MEASurement:INSPect:DETEctor<x> . . . 2087
SENSe:EMC:MEASurement:INSPect:DETEctor<x>:COMParision:LEVEl . . . 2088
SENSe:EMC:MEASurement:INSPect:DETEctor<x>:ENABle . . . 2089
SENSe:EMC:MEASurement:INSPect:DWELl:TIME . . . 2089
SENSe:EMC:MEASurement:INSPect:DWELl:TIME:AUTO . . . 2090
SENSe:EMC:MEASurement:INSPect:MODE . . . 2090
SENSe:EMC:MEASurement:INSPect:MODE:DISCrete: FREQuency<x>:ENABle . . . 2091
SENSe:EMC:MEASurement:INSPect:MODE:DISCrete: FREQuency<x>:FREQuency . . . 2091
SENSe:EMC:MEASurement:INSPect:MODE:DISCrete: FREQuency<x>:RBW . . . 2092
SENSe:EMC:MEASurement:INSPect:MODE:DISCrete: FREQuency<x>:RBW:AUTO . . . 2092
SENSe:EMC:MEASurement:INSPect:MODE:DISCrete:LOAD: EMISsion:RESults (No Query Form) . . . 2093
SENSe:EMC:MEASurement:INSPect:MODE:HARMonic<x>:ENABle . . . 2093
SENSe:EMC:MEASurement:INSPect:MODE:HARMonic<x>: FACTor<y>:ENABle . . . 2094
SENSe:EMC:MEASurement:INSPect:MODE: HARMonic<x>:FACTor<y>:FREQuency . . . 2094
SENSe:EMC:MEASurement:INSPect:MODE: HARMonic<x>:FACTor<y>:HARMonic:FACTor . . . 2095
SENSe:EMC:MEASurement:INSPect:MODE: HARMonic<x>:FACTor<y>:RBW . . . 2096
SENSe:EMC:MEASurement:INSPect:MODE:HARMonic<x>: FACTor<y>:RBW:AUTO . . . 2096

SENSe:EMC:MEASurement:INSPect:MODE:HARMonic<x>:FREquency 2097
SENSe:EMC:MEASurement:INSPect:STARt (No Query Form) 2097
SENSe:EMC:MEASurement:SCAN:DETEctor<x> 2098
SENSe:EMC:MEASurement:SCAN:DETEctor<x>:ENABle 2099
SENSe:EMC:MEASurement:SCAN:DETEctor<x>:LIMit 2099
SENSe:EMC:MEASurement:SCAN:DWELl:TIME 2100
SENSe:EMC:MEASurement:SCAN:DWELl:TIME:AUTO 2100
SENSe:EMC:MEASurement:SCAN:STARt (No Query Form) 2101
SENSe:EMC:MEASurement:SPOT:ALL 2101
SENSe:EMC:MEASurement:SPOT:COUNt 2102
SENSe:EMC:MEASurement:SPOT:DETEctor<x> 2102
SENSe:EMC:MEASurement:SPOT:DETEctor<x>:ENABle 2103
SENSe:EMC:MEASurement:SPOT:DETEctor<x>:LIMit 2103
SENSe:EMC:MEASurement:SPOT:DWELl:TIME 2104
SENSe:EMC:MEASurement:SPOT:DWELl:TIME:AUTO 2105
SENSe:EMC:MEASurement:SPOT:STARt (No Query Form) 2105
SENSe:EMC:MEASurement:TYPE 2105
SENSe:EMC:MODE 2106
SENSe:EMC:OPTimization 2106
SENSe:EMC:POINts:COUNt 2107
SENSe:EMC:RANGe<x>:BANDwidth:VIDeo 2108
SENSe:EMC:RANGe<x>:BANDwidth:VIDeo:STATe 2108
SENSe:EMC:RANGe<x>:EXCursion 2109
SENSe:EMC:RANGe<x>:FILTer[:SHAPe]:BANDwidth 2109
SENSe:EMC:RANGe<x>:FILTer[:SHAPe]:BANDwidth:AUTO 2110
SENSe:EMC:RANGe<x>:FREQuency:STARt 2110
SENSe:EMC:RANGe<x>:FREQuency:STOP 2111
SENSe:EMC:RANGe<x>:LIMit<y>:EQUAl 2111
SENSe:EMC:RANGe<x>:LIMit<y>:STARt 2112
SENSe:EMC:RANGe<x>:LIMit<y>:STOP 2112
SENSe:EMC:RANGe<x>:STATe 2113
SENSe:EMC:RANGe<x>:THReshold 2113
SENSe:EMC:REGion 2114
SENSe:EMC:REPOrt:INCLude:CONTrol:SETTings 2114
SENSe:EMC:REPOrt:INCLude:EMISsion:RESUlts 2115
SENSe:EMC:REPOrt:INCLude:ENVIronment:INFOrmation 2115
SENSe:EMC:REPOrt:INCLude:ENVIronment:INFOrmation:ADD (No Query Form) 2116
SENSe:EMC:REPOrt:INCLude:ENVIronment:INFOrmation:DELete:ALL (No Query Form) 2116
SENSe:EMC:REPOrt:INCLude:EUT:INFOrmation 2117
SENSe:EMC:REPOrt:INCLude:EUT:INFOrmation:ADD (No Query Form) 2117
SENSe:EMC:REPOrt:INCLude:EUT:INFOrmation:DELete:ALL (No Query Form) 2118
SENSe:EMC:REPOrt:INCLude:GRAPh 2118
SENSe:EMC:REPOrt:INCLude:INSPect:FREQuency:RESUlts 2119
SENSe:EMC:REPOrt:INCLude:USER:INFOrmation 2119
SENSe:EMC:REPOrt:INCLude:USER:INFOrmation:ADD (No Query Form) 2120
SENSe:EMC:REPOrt:INCLude:USER:INFOrmation:DELete:ALL (No Query Form) 2120

SENSe:EMC:REPOrt:MEASurement:HEADing 2121
SENSe:EMC:STANdard 2121
[SENSe]:FDVTime[:MEASview<y>]:SELEcted:SOURce 2122
[SENSe]:{FM|PM}:BURSt:THReshold 2122
[SENSe]:{FM|PM}:FREQuency:OFFSet 2123
[SENSe]:{FM|PM}:FREQuency:OFFSet:MARKer (No Query Form) 2123
[SENSe]:{FM|PM}:FREQuency:SEARch:AUTO 2124
[SENSe]:{FSETtling|PSETtling}:{BANDwidth|BWIDTH} 2124
[SENSe]:{FSETtling|PSETtling}:{BANDwidth|BWIDTH}:ACTual? (Query Only) 2125
[SENSe]:{FSETtling|PSETtling}:FREQuency:CENTer 2125
[SENSe]:{FSETtling|PSETtling}:FREQuency:OFFSet 2126
[SENSe]:{FSETtling|PSETtling}:LENGth 2126
[SENSe]:{FSETtling|PSETtling}:LENGth:ACTual? (Query Only) 2126
[SENSe]:{FSETtling|PSETtling}:MASK:LIMit 2127
[SENSe]:{FSETtling|PSETtling}:MASK:STATe 2128
[SENSe]:{FSETtling|PSETtling}:MASK:TIME:REFerence 2128
[SENSe]:{FSETtling|PSETtling}:MASK:TIME:STARt 2128
[SENSe]:{FSETtling|PSETtling}:MASK:TIME:STOP 2129
[SENSe]:{FSETtling|PSETtling}:MAXTracepoints 2129
[SENSe]:{FSETtling|PSETtling}:SDURation:MINimum 2130
[SENSe]:{FSETtling|PSETtling}:TARGet:REFerence 2130
[SENSe]:{FSETtling|PSETtling}:TOLerance 2131
[SENSe]:FVTime[:MEASview<y>]:CLEar:RESults (No Query Form) 2131
[SENSe]:FVTime[:MEASview<y>]:COLor 2132
[SENSe]:FVTime[:MEASview<y>]:COLor:MAXimum 2132
[SENSe]:FVTime[:MEASview<y>]:COLor:MINimum 2133
[SENSe]:FVTime[:MEASview<y>]:COLor:REVerse[:STATe] 2133
[SENSe]:FVTime[:MEASview<y>]:FREQuency:CENTer 2134
[SENSe]:FVTime[:MEASview<y>]:FREQuency:SPAN 2135
[SENSe]:FVTime[:MEASview<y>]:FREQuency:SPAN:LINK 2135
[SENSe]:FVTime[:MEASview<y>]:FREQuency:STARt 2136
[SENSe]:FVTime[:MEASview<y>]:FREQuency:STEP 2136
[SENSe]:FVTime[:MEASview<y>]:FREQuency:STEP:AUTO 2137
[SENSe]:FVTime[:MEASview<y>]:FREQuency:STOP 2137
[SENSe]:FVTime[:MEASview<y>]:MAXTracepoints 2138
[SENSe]:FVTime[:MEASview<y>]:OGRAm:COLor 2138
[SENSe]:FVTime[:MEASview<y>]:OGRAm:COLor:MAXimum 2139
[SENSe]:FVTime[:MEASview<y>]:OGRAm:COLor:MINimum 2140
[SENSe]:FVTime[:MEASview<y>]:OGRAm:COLor:REVerse 2141
[SENSe]:FVTime[:MEASview<y>]:PLOT 2141
[SENSe]:FVTime[:MEASview<y>]:SELEcted:SOURce 2142
[SENSe]:IQVTime[:MEASview<y>]:CLEar:RESults (No Query Form) 2142
[SENSe]:IQVTime[:MEASview<y>]:FREQuency:CENTer 2143
[SENSe]:IQVTime[:MEASview<y>]:FREQuency:SPAN 2143
[SENSe]:IQVTime[:MEASview<y>]:FREQuency:SPAN:LINK 2144
[SENSe]:IQVTime[:MEASview<y>]:FREQuency:STARt 2144

[SENSe]:IQVTime[:MEASview<y>]:FREQuency:STEP 2145
[SENSe]:IQVTime[:MEASview<y>]:FREQuency:STEP:AUTO 2146
[SENSe]:IQVTime[:MEASview<y>]:FREQuency:STOP 2146
[SENSe]:IQVTime[:MEASview<y>]:MAXTracepoints 2147
[SENSe]:IQVTime[:MEASview<y>]:SELEcted:SOURce 2147
SENSe:LTE:ACLR:BANDwidth:RESolution 2148
SENSe:LTE:ACLR:BANDwidth:VIDeo 2148
SENSe:LTE:ACLR:BANDwidth:VIDeo:STATe 2148
SENSe:LTE:ACLR:CHANnel:IBANdwidth 2149
SENSe:LTE:ACLR:CHANnel:WIDTh 2149
SENSe:LTE:ACLR:CLEar:RESults (No Query Form) 2150
SENSe:LTE:ACLR:FREQuency 2150
SENSe:LTE:ACLR:MFILter 2151
SENSe:LTE:ACLR:MODE:REALtime 2151
SENSe:LTE:ACLR:NFLoor:ACQuire (No Query Form) 2152
SENSe:LTE:ACLR:NFLoor:CORRected? (Query Only) 2152
SENSe:LTE:ACLR:NFLoor:STATe 2153
SENSe:LTE:ACLR:OFFSet<x>:BANDwidth:RESolution 2153
SENSe:LTE:ACLR:OFFSet<x>:BANDwidth:RESolution:COUNt 2154
SENSe:LTE:ACLR:OFFSet<X>:BANDwidth:RESolution:FILTer 2154
SENSe:LTE:ACLR:OFFSet<x>:BANDwidth:RESolution:VIDeo 2155
SENSe:LTE:ACLR:OFFSet<x>:BANDwidth:RESolution:VIDeo:STATe 2155
SENSe:LTE:ACLR:OFFSet<x>:DETection 2156
SENSe:LTE:ACLR:OFFSet<x>:FREQuency:BANDwidth 2156
SENSe:LTE:ACLR:OFFSet<x>:FREQuency:OFFSet 2157
SENSe:LTE:ACLR:OFFSet<x>:LIMit:ABSolute 2157
SENSe:LTE:ACLR:OFFSet<x>:LIMit:MASK 2158
SENSe:LTE:ACLR:OFFSet<x>:LIMit:RELative 2158
SENSe:LTE:ACLR:OFFSet<x>:LIMit:SIDE 2159
SENSe:LTE:ACLR:OFFSet<x>:STATe 2159
SENSe:LTE:ACLR:POINts:COUNt 2160
SENSe:LTE:ACLR:REFerence:MAGNitude 2160
SENSe:LTE:ACLR:REFerence:MAGNitude:AUTO 2161
SENSe:LTE:ACLR:REFerence:MAGNitude:DETection 2161
SENSe:LTE:ACLR:STEP:AUTO 2162
SENSe:LTE:ACLR:STEP:MAGNitude 2162
SENSe:LTE:ANALysis:AVErage:OVER:OFFSlot:STATe 2163
SENSe]:LTE:ANALysis:EQUAlization:STATe 2163
SENSe:LTE:ANALysis:LENGth 2164
SENSe:LTE:ANALysis:LENGth:ACTual? (Query Only) 2164
SENSe:LTE:ANALysis:LENGth:AUTO 2165
[SENSe]:LTE:ANALysis:OFFSet 2165
SENSe:LTE:ANALysis:OFFSet:AUTO 2166
SENSe:LTE:ANALysis:TIMe:UNIT 2166
SENSe:LTE:ANTEnna:PORTs:CHOSen 2167
SENSe:LTE:ANTEnna:PORTs:TOTal 2167

SENSe:LTE:AUTO:DETect 2168
SENSe:LTE:CHANnel:(BANDwidth|BWIDth) 2168
SENSe:LTE:CHANnel:BANDwidth 2169
SENSe:LTE:CHSPectrum:AVERage 2169
SENSe:LTE:CHSPectrum:AVERage:COUNt 2170
SENSe:LTE:CHSPectrum:BANDwidth:RESolution 2170
SENSe:LTE:CHSPectrum:BANDwidth:RESOlution:AUTO 2171
SENSe:LTE:CHSPectrum:BANDwidth:VIDeo 2171
SENSe:LTE:CHSPectrum:BANDwidth:VIDeo:STATe 2172
SENSe:LTE:CHSPectrum:CHANnel:BANDwidth 2172
SENSe:LTE:CHSPectrum:CLEar:RESults (No Query Form) 2172
SENSe:LTE:CHSPectrum:DETection 2173
SENSe:LTE:CHSPectrum:FREQuency 2173
SENSe:LTE:CHSPectrum:FREQuency:SPAN 2174
SENSe:LTE:CHSPectrum:FREQuency:STEP 2174
SENSe:LTE:CHSPectrum:FREQuency:STEP:AUTO 2175
SENSe:LTE:CHSPectrum:POINts:COUNt 2175
SENSe:LTE:CYClic:PREFix 2176
SENSe:LTE:FRAMe:STRUcture 2176
SENSe:LTE:LINK:CONFiguration 2177
SENSe:LTE:SUBFrame:CONFiguration 2177
[SENSe]:MCPower:AVERage 2178
[SENSe]:MCPower:AVERage:COUNt 2178
[SENSe]:MCPower:{BANDwidth|BWIDth}[:RESolution] 2179
[SENSe]:MCPower:{BANDwidth|BWIDth}[:RESolution]:ACTual? (Query Only) 2179
[SENSe]:MCPower:{BANDwidth|BWIDth}[:RESolution]:AUTO 2180
[SENSe]:MCPower:{BANDwidth|BWIDth}:VIDeo 2180
[SENSe]:MCPower:{BANDwidth|BWIDth}:VIDeo:STATe 2180
[SENSe]:MCPower:CHANnel:ADJacent:ADD (No Query Form) 2181
[SENSe]:MCPower:CHANnel:ADJacent:DELete (No Query Form) 2182
[SENSe]:MCPower:CHANnel:ADJacent:LIMIts<x> 2182
[SENSe]:MCPower:CHANnel:ADJacent:LIMIts<x>:ENABle? (Query Only) 2182
[SENSe]:MCPower:CHANnel:FILTer 2183
[SENSe]:MCPower:CHANnel:MAIN:{BANDwidth|BWIDth} 2183
[SENSe]:MCPower:CHANnel:MAIN:COUNt 2184
[SENSe]:MCPower:CHANnel:MAIN:INACtive 2184
[SENSe]:MCPower:CHANnel:MAIN:SPACing 2185
[SENSe]:MCPower:CHIPrate 2185
[SENSe]:MCPower:CLEar:RESults (No Query Form) 2185
[SENSe]:MCPower:FREQuency 2186
[SENSe]:MCPower:FREQuency:STEP 2186
[SENSe]:MCPower:FREQuency:STEP:AUTO 2187
[SENSe]:MCPower:NFLoor:STATe 2187
[SENSe]:MCPower:POINts:COUNt 2188
[SENSe]:MCPower:RCHannels? (Query Only) 2188
[SENSe]:MCPower:RCHannels:MAIN<x> (No Query Form) 2189

[SENSe]:MCPower:RCHannels:TOTal (No Query Form) 2189
[SENSe]:MCPower:RRCRolloff 2190
[SENSe]:MEASurement:FREQuency[:SOURce<z>] 2190
[SENSe]:MEASurement:FREQuency[:SOURce<z>]:CENTer:LOCK 2191
[SENSe]:MERRor[:MEASview<y>]:SELEcted:SOURce 2191
[SENSe]:NOISe:AVERage:COUNt 2192
[SENSe]:NOISe:BANDwidth 2192
[SENSE]:NOISe:CONTrol:SETTling:OFF[:TIME] 2193
[SENSE]:NOISe:CONTrol:SETTling:ON[:TIME] 2193
[SENSe]:NOISe:CORRection:ENR:CONSTant 2194
[SENSe]:NOISe:CORRection:ENR:MODE 2194
[SENSe]:NOISe:CORRection:ENR:TABLe:DATA? (Query Only) 2195
[SENSe]:NOISe:CORRection:ENR:TABLe:ID 2195
[SENSe]:NOISe:CORRection:ENR:TABLe:NEW (No Query Form) 2196
[SENSe]:NOISe:CORRection:ENR:TABLe:SERial 2197
[SENSe]:NOISe:CORRection:TCOLd:ADUT:UREFerence 2197
[SENSe]:NOISe:CORRection:TCOLd:ADUT[:VALue] 2198
[SENSe]:NOISe:CORRection:TCOLd:BDUT:UREFerence 2198
[SENSe]:NOISe:CORRection:TCOLd:BDUT[:VALue] 2199
[SENSe]:NOISe:CORRection:TCOLd[:REFerence] 2199
[SENSe]:NOISe:FIGure:POINTs:COUNt (No Query Form) 2200
[SENSe]:NOISe:FREQuency:CENTer 2200
[SENSe]:NOISe:FREQuency:MODE 2201
[SENSe]:NOISe:FREQuency:SPAN 2201
[SENSe]:NOISe:FREQuency:STARt 2202
[SENSe]:NOISe:FREQuency:STEPs 2202
[SENSe]:NOISe:FREQuency:STOP 2203
[SENSe]:NOISe:FREQuency:TABLe:DATA? (Query Only) 2203
[SENSe]:NOISe:FREQuency:TABLe:NEW (No Query Form) 2204
[SENSe]:OBWidth:AVERage 2204
[SENSe]:OBWidth:AVERage:COUNt 2204
[SENSe]:OBWidth:{BANDwidth|BWIDth}:MEASurement 2205
[SENSe]:OBWidth:{BANDwidth|BWIDth}[:RESolution] 2205
[SENSe]:OBWidth:{BANDwidth|BWIDth}[:RESolution]:ACTual? (Query Only) 2206
[SENSe]:OBWidth:{BANDwidth|BWIDth}[:RESolution]:AUTO 2206
[SENSe]:OBWidth:{BANDwidth|BWIDth}:VIDeo 2207
[SENSe]:OBWidth:{BANDwidth|BWIDth}:VIDeo:STATe 2207
[SENSe]:OBWidth:BOBW:XDBM:MEASurement:DIRection 2208
[SENSe]:OBWidth:BOBW:XDBM:RANGe 2208
[SENSe]:OBWidth:CLEar:RESults (No Query Form) 2208
[SENSe]:OBWidth:FREQuency:CENTer 2209
[SENSe]:OBWidth:FREQuency:STEP 2209
[SENSe]:OBWidth:FREQuency:STEP:AUTO 2210
[SENSe]:OBWidth:PERCent 2210
[SENSe]:OBWidth:XDBLevel 2211
[SENSe]:OFDM:CHANnel[:BANDwidth|BWIDth] 2211

[SENSe]:OFDM:CHANnel[:BANDwidth|BWIDth]:SRATe? (Query Only) 2212
[SENSe]:OFDM:CHANnel[:BANDwidth|BWIDth]:STATe 2212
[SENSe]:OFDM:CHANnel:ESTimation 2213
[SENSe]:OFDM:CONSte:DETermination 2213
[SENSe]:OFDM:CONSte:DETermination:MANual 2213
[SENSe]:OFDM:GUARd:INTerval 2214
[SENSe]:OFDM:GUARd:INTerval:USER 2214
[SENSe]:OFDM:PILot:TRACking:AMPLitude:STATe 2215
[SENSe]:OFDM:PILot:TRACking:PHASe:STATe 2215
[SENSe]:OFDM:PILot:TRACking:TIMing:STATe 2216
[SENSe]:OFDM:RADix (No Query Form) 2216
[SENSe]:OFDM:SCARrier:SPACing 2217
[SENSe]:OFDM:SCARrier:SPACing:STATe 2217
[SENSe]:OFDM:SCARriers 2217
[SENSe]:OFDM:SCARriers:RANGe:STARt 2218
[SENSe]:OFDM:SCARriers:RANGe:STOP 2218
[SENSe]:OFDM:SCARriers:SINGle:INDex 2219
[SENSe]:OFDM:FFT:LENGth? (Query Only) 2219
[SENSe]:OFDM:STANdard 2220
[SENSe]:OFDM:SWAP:IQ 2220
[SENSe]:OFDM:SYMBol:ANALysis:OFFSet 2220
[SENSe]:OFDM:SYMBols 2221
[SENSe]:OFDM:SYMBols:MAX 2221
[SENSe]:OFDM:SYMBols:MAX:STATe 2222
[SENSe]:OFDM:SYMBols:RANGe:COUNt 2222
[SENSe]:OFDM:SYMBols:RANGe:STARt 2223
[SENSe]:OFDM:SYMBols:SINGle:INDex 2223
[SENSe]:OFDM:UNIT:FREQuency (No Query Form) 2223
[SENSe]:OFDM:UNIT:TIME (No Query Form) 2224
[SENSe]:P25:ANALysis:OFFSet 2224
[SENSe]:P25:ANALysis:OFFSet:AUTO 2225
[SENSe]:P25([:BANDwidth]|:BWIDth):MODE 2225
[SENSe]:P25([:BANDwidth]|:BWIDth):TINTerval 2226
[SENSe]:P25([:BANDwidth]|:BWIDth):TINTerval:AUTO 2226
[SENSe]:P25:FILTer:MEASurement 2227
[SENSe]:P25:FILTer:REFerence 2228
[SENSe]:P25:FREQuency:ERRor 2229
[SENSe]:P25:FREQuency:ERRor:AUTO 2229
[SENSe]:P25:MODUlation:CHANnel:([BANDwidth]|:BWIDth)]? (Query Only) 2230
[SENSe]:P25:MODUlation:STANdard 2230
SENSe:P25:MODUlation:TYPE 2231
[SENSe]:P25:SRATe 2231
[SENSe]:P25:SUMMary:CLEar:RESults (No Query Form) 2232
[SENSe]:P25:SUMMary:LIMIt:FA 2232
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:AVERage: TERRor:ONE 2233
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:AVERage: TERRor:ZERO 2233

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:LOGIc:CHANnel: OFF:SLOT 2234
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:LOGIc:CHANnel: PEAK:ACPR 2234
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PEAK: NEGAtive:LO 2235
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PEAK: NEGAtive:UP 2236
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PEAK: POSItive:LO 2236
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PEAK: POSItive:UP 2237
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation: PHASe[1/2]:F[1/2/3/4]:LO 2237
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PHASe[1/2]: F[1/2/3/4]:LO 2238
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PHASe[1/2]: F[1/2/3/4]:UP 2238
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PHASe[1/2]: F[1/2/3/4]:UP 2239
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMAX:OFF 2239
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMAX:OFF 2240
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMAX:ON 2240
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMAX:ON 2241
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMAX:SS 2241
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMIN:SS 2242
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter: AVERage:ATTK:ENCOder 2242
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter: AVERage:ATTK:ENCOder:BI 2243
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter: AVERage:ATTK:POWEr 2243
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter: AVERage:ATTK:POWEr:BI 2244
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter: AVERage:THRUput:DLAY 2245
[SENSe]:P25:SUMMary:LIMIt:MF 2245
[SENSe]:P25:SUMMary:LIMIt:RF:OUTPower 2246
[SENSe]:P25:SUMMary:LIMIt:SRA 2246
[SENSe]:P25:SYMBols:POINts 2247
[SENSe]:P25:TEST:PATTern 2247
[SENSe]:P25:TIMe:UNITs 2248
[SENSe]:P25:TRIGgger:MEASurement 2248
[SENSe]:P25:TRIGgger:MEASurement:TOBSync 2249
[SENSe]:P25:TRIGgger:MEASurement:TOBSync:AUTO 2249
[SENSe]:P25:TRIGgger:MEASurement:TRANsient: FREQuency:BEHAviour 2250
[SENSe]:P25:TRIGgger:MEASurement:TRANsient:FREQuency: BEHAviour:TIME:OFF 2250
[SENSe]:P25:TRIGgger:MEASurement:TRANsient:FREQuency: BEHAviour:TIME:OFF:AUTO 2251
[SENSe]:P25:TRIGgger:MEASurement:TRANsient:FREQuency: BEHAviour:TIME:ON 2251
[SENSe]:P25:TRIGgger:MEASurement:TRANsient: FREQuency:BEHAviour:TIME:ON:AUTO 2252
[SENSe]:PERRor[:MEASview<y>]:SELEcted:SOURce 2252
[SENSe]:PHVTime[:MEASview<y>]:CLEar:RESults (No Query Form) 2253
[SENSe]:PHVTime[:MEASview<y>]:COLor 2253
[SENSe]:PHVTime[:MEASview<y>]:COLor:MAXimum 2254
[SENSe]:PHVTime[:MEASview<y>]:COLor:MINimum 2255
[SENSe]:PHVTime[:MEASview<y>]:COLor:REVerse[:STATe] 2255
[SENSe]:PHVTime[:MEASview<y>]:FREQuency:CENTer 2256
[SENSe]:PHVTime[:MEASview<y>]:FREQuency:SPAN 2256
[SENSe]:PHVTime[:MEASview<y>]:FREQuency:SPAN:LINK 2257
[SENSe]:PHVTime[:MEASview<y>]:FREQuency:STARt 2257
[SENSe]:PHVTime[:MEASview<y>]:FREQuency:STEP 2258

[SENSe]:PHVTime[:MEASview<y>]:FREQuency:STEP:AUTO 2258
[SENSe]:PHVTime[:MEASview<y>]:FREQuency:STOP 2259
[SENSe]:PHVTime[:MEASview<y>]:MAXTracepoints 2259
[SENSe]:PHVTime[:MEASview<y>]:OGRAm:COLor 2260
[SENSe]:PHVTime[:MEASview<y>]:OGRAm:COLor:MAXimum 2261
[SENSe]:PHVTime[:MEASview<y>]:OGRAm:COLor:MINimum 2262
[SENSe]:PHVTime[:MEASview<y>]:OGRAm:COLor:REVerse 2262
[SENSe]:PHVTime[:MEASview<y>]:PLOT 2263
[SENSe]:PHVTime[:MEASview<y>]:SELEcted:SOURce 2263
[SENSe]:PM:PHASe:OFFSet 2264
[SENSe]:PM:PHASe:OFFSet:MARKer (No Query Form) 2264
[SENSe]:PM:PHASe:SEARch:AUTO 2265
[SENSe]:PNOise:AVERage:COUNt 2265
[SENSe]:PNOise:AVERage:ENABle 2266
[SENSe]:PNOise:CARRier:FREQuency:TRACk 2266
[SENSe]:PNOise:CARRier:THReshold 2266
[SENSe]:PNOise:CLEar:RESults (No Query Form) 2267
[SENSe]:PNOise:FREQuency:INTegration:OFFSet:STARt 2267
[SENSe]:PNOise:FREQuency:INTegration:OFFSet:STOP 2268
[SENSe]:PNOise:FREQuency:PLOT:OFFSet:STARt 2268
[SENSe]:PNOise:FREQuency:PLOT:OFFSet:STOP 2268
[SENSe]:PNOise:OPTimization 2269
[SENSe]:POWer:UNITs 2269
[SENSe]:PULSe[:MEASview<y>]:ANALyze:FDOMain:MEASurement: TIME:ALENgth 2270
[SENSe]:PULSe[:MEASview<y>]:ANALyze:FDOMain:MEASurement: TIME:METHod 2271
[SENSe]:PULSe[:MEASview<y>]:ANALyze:FDOMain:MEASurement: TIME:RLENgth 2271
[SENSe]:PULSe[:MEASview<y>]:ANALyze:FDOMain:MEASurement: TIME:RLEVel 2272
[SENSe]:PULSe[:MEASview<y>]:ANALyze:FDOMain:MEASurement: TIME:STARt 2273
[SENSe]:PULSe[:MEASview<y>]:ANALyze:IRESponse:CORRection: AMPLitude[:STATe] 2273
[SENSe]:PULSe[:MEASview<y>]:ANALyze:IRESponse:KOTime 2274
[SENSe]:PULSe[:MEASview<y>]:ANALyze:IRESponse:MEASurement: TIME:ALENgth 2274
[SENSe]:PULSe[:MEASview<y>]:ANALyze:IRESponse:MEASurement: TIME: METHod 2275
[SENSe]:PULSe[:MEASview<y>]:ANALyze:IRESponse:MEASurement: TIME:RLENgth 2276
[SENSe]:PULSe[:MEASview<y>]:ANALyze:IRESponse:MEASurement: TIME: RLEVel 2276
[SENSe]:PULSe[:MEASview<y>]:ANALyze:IRESponse:MEASurement: TIME:STARt 2277
[SENSe]:PULSe[:MEASview<y>]:ANALyze: LEVel 2277
[SENSe]:PULSe[:MEASview<y>]:ANALyze:LEVel:FIFTy 2278
[SENSe]:PULSe[:MEASview<y>]:ANALyze:LEVel:HUNDred 2278
[SENSe]:PULSe[:MEASview<y>]:ANALyze:MEASurement:TIME:TYPE? 2279
[SENSe]:PULSe[:MEASview<y>]:ANALyze:MEASurement:TIME:TYPE 2280
[SENSe]:PULSe[:MEASview<y>]:ANALyze:OVERshoot: MEASurement:TIME:ALENgth 2280
[SENSe]:PULSe[:MEASview<y>]:ANALyze:OVERshoot: MEASurement:TIME: METHod 2281
[SENSe]:PULSe[:MEASview<y>]:ANALyze:OVERshoot: MEASurement:TIME:RLENgth 2281
[SENSe]:PULSe[:MEASview<y>]:ANALyze:OVERshoot: MEASurement:TIME: RLEVel 2282
[SENSe]:PULSe[:MEASview<y>]:ANALyze:OVERshoot: MEASurement:TIME: STARt 2283
[SENSe]:PULSe[:MEASview<y>]:ANALyze:PMLocation 2283

[SENSe]:PULSe[:MEASview<y>]:ANALyze:POINt:LOCation 2284
[SENSe]:PULSe[:MEASview<y>]:ANALyze:RFALl 2284
[SENSe]:PULSe[:MEASview<y>]:ANALyze:RIPPle 2285
[SENSe]:PULSe:CARRier:FREQuency? (Query Only) 2285
[SENSe]:PULSe[:MEASview<y>]:CARRier:OFFSet 2286
[SENSe]:PULSe[:MEASview<y>]:CARRier:SEARch 2287
[SENSe]:PULSe:CUMulative:HISTogram[:MEASview<y>]:BIN:COUNt 2287
[SENSe]:PULSe:CUMulative:HISTogram[:MEASview<y>]:CLEar: RESults (Query Only) 2288
[SENSe]:PULSe:CUMulative:HISTogram[:MEASview<y>]:MODE 2288
[SENSe]:PULSe:CUMulative:HISTogram[:MEASview<y>]:ORDinate (Query Only) 2289
[SENSe]:PULSe:CUMulative:HISTogram[:MEASview<y>]: OUTLier:ORDinate 2289
[SENSe]:PULSe:CUMulative:HISTogram[:MEASview<y>]: SETtings:APPly (No Query Form) 2290
[SENSe]:PULSe:CUMulative:HISTogram[:MEASview<y>]:XLIMit 2290
SENSe:PULSe:CUMulative:HISTogram[:MEASview<y>]: SELEcted:SOURce 2291
[SENSe]:PULSe:CUMulative:STATistics[:MEASview<y>]:CLEar: RESults 2291
[SENSe]:PULSe:CUMulative:STATistics[:MEASview<y>]: SELEcted:SOURce 2292
[SENSe]:PULSe[:MEASview<y>]:DETect:MEASurement 2292
[SENSe]:PULSe[:MEASview<y>]:DETect:NUMBer 2293
[SENSe]:PULSe[:MEASview<y>]:DETect:NUMBer:THReshold 2293
[SENSe]:PULSe[:MEASview<y>]:DETect:TIMe[:THReshold] 2294
[SENSe]:PULSe[:MEASview<y>]:FILTer:{BANDwidth|BWIDth} 2295
[SENSe]:PULSe[:MEASview<y>]:FILTer:MEASurement 2295
[SENSe]:PULSe[:MEASview<y>]:FREFerence:AUTO 2296
[SENSe]:PULSe[:MEASview<y>]:FREFerence:CSLope 2297
[SENSe]:PULSe[:MEASview<y>]:FREFerence:CSLope:AUTO 2297
[SENSe]:PULSe[:MEASview<y>]:FREFerence:OFFSet 2298
[SENSe]:PULSe[:MEASview<y>]:MODulation:TYPE 2298
[SENSe]:PULSe[:MEASview<y>]:STATistics:SELEcted:SOURce 2299
[SENSe]:PULSe:OGRAm[:MEASview<y>]: ANALysis:FVTime:FILTer:SHAPe 2299
[SENSe]:PULSe:OGRAm[:MEASview<y>]:ANALysis:FVTime: SPECtrum:ANALysis:REGion 2300
[SENSe]:PULSe:OGRAm[:MEASview<y>]:SELEcted:SOURce 2301
[SENSe]:PULSe:OGRAm[:MEASview<y>]:ANALysis:FVTime: ONTime:LENgth 2301
[SENSe]:PULSe:OGRAm[:MEASview<y>]:ANALysis:FVTime: INDependent:STARt 2302
[SENSe]:PULSe:OGRAm[:MEASview<y>]:ANALysis:FVTime: INDependent:LENgth 2302
[SENSe]:PULSe[:MEASview<y>]:SIGNal:TYPE 2303
[SENSe]:PULSe[:MEASview<y>]:STATistics:HISTogram:ORDinate 2304
[SENSe]:PULSe[:MEASview<y>]:STATistics:HISTogram:POINts 2304
[SENSe]:PULSe[:MEASview<y>]:STATistics:HISTogram:POINts:AUTO 2305
[SENSe]:PULSe[:MEASview<y>]:STATistics:IN:LINear:SCALe 2305
[SENSe]:PULSe:TRACe[:MEASview<y>]:SELEcted:SOURce 2306
[SENSe]:PULSe:RESult[:MEASview<y>]:SELEcted:SOURce 2306
[SENSe]:REANalyze (No Query Form) 2307
[SENSe]:REANalyze:ACQuisition:SETTings? (Query Only) 2307
[SENSe]:REANalyze:ALL:LOOP (No Query Form) 2307
[SENSe]:REANalyze:CURRent (No Query Form) 2308
[SENSe]:REANalyze:CURRent:ACQuisition? (Query Only) 2308

[SENSe]:REANalyze:CURRent:FRAMe? (Query Only) 2308
[SENSe]:REANalyze:CURRent (No Query Form) 2308
[SENSe]:REANalyze:CURRent:TIMestamp? (Query Only) 2309
[SENSe]:REANalyze:DELete:DATA (No Query Form) 2309
[SENSe]:REANalyze:FIRSt (No Query Form) 2309
[SENSe]:REANalyze:LAST (No Query Form) 2309
[SENSe]:REANalyze:NEXT (No Query Form) 2310
[SENSe]:REANalyze:PAUSe 2310
[SENSe]:REANalyze:PREVious (No Query Form) 2310
[SENSe]:REANalyze:SELect:ACQuisition:DATA:TYPE 2311
[SENSe]:REANalyze:SELect:ACQuisition:FIRSt? (Query Only) 2311
[SENSe]:REANalyze:SELect:ACQuisition:FIRSt:TIMestamp? (Query Only) 2311
[SENSe]:REANalyze:SELect:ACQuisition:LAST? (Query Only) 2312
[SENSe]:REANalyze:SELect:ACQuisition:LAST:TIMestamp? (Query Only) 2312
[SENSe]:REANalyze:SELect:ACQuisition:STARt 2312
[SENSe]:REANalyze:SELect:ACQuisition:STARt:TIMestamp? (Query only) 2313
[SENSe]:REANalyze:SELect:ACQuisition:STOP 2313
[SENSe]:REANalyze:SELect:ACQuisition:STOP:TIMestamp? (Query Only) 2313
[SENSe]:REANalyze:SELect:ALL (No Query Form) 2314
[SENSe]:REANalyze:SELect:FRAMe:FIRSt? (Query Only) 2314
[SENSe]:REANalyze:SELect:FRAMe:FIRSt:TIMestamp? (Query Only) 2314
[SENSe]:REANalyze:SELect:FRAMe:LAST? (Query Only) 2315
[SENSe]:REANalyze:SELect:FRAMe:LAST:TIMestamp? (Query Only) 2315
[SENSe]:REANalyze:SELect:FRAMe:STARt 2315
[SENSe]:REANalyze:SELect:FRAMe:STARt:TIMestamp? (Query Only) 2316
[SENSe]:REANalyze:SELect:FRAMe:STOP 2316
[SENSe]:REANalyze:SELect:FRAMe:STOP:TIMestamp? (Query Only) 2316
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:ALL (No Query Form) 2317
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:CURRent: ACQuisition? (Query Only) 2317
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:CURRent: TIMestamp? (Query Only) 2317
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:FIRSt? (Query Only) 2318
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:FIRSt: TIMestamp? (Query Only) 2318
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:LAST? (Query Only) 2319
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:LAST: TIMestamp? (Query Only) 2319
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:STARt 2319
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:STARt: TIMestamp? (Query Only) 2320
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:STOP 2320
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:STOP: TIMestamp? (Query Only) 2321
[SENSe]:REANalyze:SPEed 2321
[SENSe]:REANalyze:STOP (No Query Form) 2321
[SENSe]:REANalyze:TIMestamp:DECimal 2322
SENSe:ROSCillator:EXTernal:TIMe 2322
SENSe:ROSCillator:EXTernal:TIMe:STATus? (Query Only) 2323
[SENSe]:ROSCillator:SOURce 2323
[SENSe]:SEM:{BANDwidth|BWIDth}[:RESolution] 2324
[SENSe]:SEM:{BANDwidth|BWIDth}:VIDeo 2324

[SENSe]:SEM:{BANDwidth|BWIDth}:VIDeo:STATe 2325
[SENSe]:SEM:CHANnel:FILTer 2325
[SENSe]:SEM:CHANnel:IBANdwidth 2325
[SENSe]:SEM:CHANnel:WIDTh 2326
[SENSe]:SEM:CHIPrate 2326
[SENSe]:SEM:CLEar:RESults (No Query Form) 2327
[SENSe]:SEM:FREQuency 2327
[SENSe]:SEM:FREQuency:SPAN? (Query Only) 2328
[SENSe]:SEM:MFILter 2328
[SENSe]:SEM:MODE:REALtime 2328
[SENSe]:SEM:MTYPe 2329
[SENSe]:SEM:OFFSet:DETection 2329
[SENSe]:SEM:OFFSet<x>:BANDwidth:FILTer? (Query Only) 2330
[SENSe]:SEM:OFFSet<x>:BANDwidth[:RESolution] 2330
[SENSe]:SEM:OFFSet<x>:BANDwidth:RESolution:COUNt 2331
[SENSe]:SEM:OFFSet<x>:BANDwidth:VIDeo 2331
[SENSe]:SEM:OFFSet<x>:BANDwidth:VIDeo:STATe 2332
[SENSe]:SEM:OFFSet<x>:FREQuency:STARt 2332
[SENSe]:SEM:OFFSet<x>:FREQuency:STOP 2333
[SENSe]:SEM:OFFSet<x>:LIMIT:ABSolute:SAME 2333
[SENSe]:SEM:OFFSet<x>:LIMIT:ABSolute:STARt 2333
[SENSe]:SEM:OFFSet<x>:LIMIT:ABSolute:STOP 2334
[SENSe]:SEM:OFFSet<x>:LIMIT:MASK 2334
[SENSe]:SEM:OFFSet<x>:LIMIT:RELative:SAME 2335
[SENSe]:SEM:OFFSet<x>:LIMIT:RELative:STARt 2335
[SENSe]:SEM:OFFSet<x>:LIMIT:RELative:STOP 2336
[SENSe]:SEM:OFFSet<x>:LIMIT:SIDE 2336
[SENSe]:SEM:OFFSet<x>:STATe (No Query Form) 2336
[SENSe]:SEM:POINts:COUNt 2337
[SENSe]:SEM:PRESet (No Query Form) 2337
[SENSe]:SEM:REFerence:AUTO 2338
[SENSe]:SEM:REFerence:DETection 2338
[SENSe]:SEM:REFerence[:MAGNitude] 2339
[SENSe]:SEM:RRCRolloff 2339
[SENSe]:SEM:STEP:AUTO 2340
[SENSe]:SEM:STEP[:MAGNitude] 2340
[SENSe]:SEM:WIDTh:DEFinition 2341
[SENSe]:SGRam[:MEASview<y>]:{BANDwidth|BWIDth} [:RESolution] 2341
[SENSe]:SGRam[:MEASview<y>]:{BANDwidth|BWIDth} [:RESolution]:ACTual (Query Only) 2342
[SENSe]:SGRam[:MEASview<y>]:{BANDwidth|BWIDth} [:RESolution]:AUTO 2342
[SENSe]:SGRam[:MEASview<y>]:{BANDwidth|BWIDth} [:RESolution]:MODE 2343
[SENSe]:SGRam[:MEASview<y>]:{BANDwidth|BWIDth}:VIDeo 2343
[SENSe]:SGRam[:MEASview<y>]:{BANDwidth|BWIDth}: VIDeo:STATe 2344
[SENSe]:SGRam[:MEASview<y>]:{BANDwidth|BWIDth}: OPTimization 2345
[SENSe]:SGRam[:MEASview<y>]:COLor 2345
[SENSe]:SGRam[:MEASview<y>]:COLor:MAXimum 2346

[SENSe]:SGRam[:MEASview<y>]:COLor:MINimum 2347
[SENSe]:SGRam[MEASview<y>]:COLor:REVerse 2347
[SENSe]:SGRam[:MEASview<y>]:CLEar:RESults (No Query Form) 2348
[SENSe]:SGRam[:MEASview<y>]:FFT:WINDow 2348
[SENSe]:SGRam[:MEASview<y>]:FILTer[:SHAPe] 2349
[SENSe]:SGRam[:MEASview<y>]:FREQuency:CENTer 2350
[SENSe]:SGRam[:MEASview<y>]:FREQuency:SPAN 2351
[SENSe]:SGRam[:MEASview<y>]:FREQuency:SPAN: BANDwidth[:RESolution]:RATio 2351
[SENSe]:SGRam[:MEASview<y>]:FREQuency:SPAN:MAXimum (No Query Form) 2352
[SENSe]:SGRam[:MEASview<y>]:FREQuency:STARt 2352
[SENSe]:SGRam[:MEASview<y>]:FREQuency:STEP 2353
[SENSe]:SGRam[:MEASview<y>]:FREQuency:STEP:AUTO 2354
[SENSe]:SGRam[:MEASview<y>]:FREQuency:STOP 2354
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:MODe 2355
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:OVERlap 2355
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:OVERlap: PERCent 2356
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:PER:DIVision 2356
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:ZOOM 2357
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:AUTO 2357
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:STARt 2358
[SENSe]:SGRam[:MEASview<y>]:SELEcted:SOURce 2358
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:PER:UPDate: MINutes 2359
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:PER:UPDate: SEConds 2359
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:SPECtrums: PERLine (Query Only) 2360
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:STARt:DIVisions 2360
[SENSe]:SGRam[:MEASview<y>]:WATerfall:DIRection 2361
[SENSe]:SGRam[:MEASview<y>]:WATerfall:ENABle 2361
[SENSe]:SGRam[:MEASview<y>]:WATerfall:Y[:SCALe]:AUTO (No Query Form) 2362
[SENSe]:SGRam[:MEASview<y>]:WATerfall:Y[:SCALe]:OFFSet 2362
[SENSe]:SGRam[:MEASview<y>]:WATerfall:Y[:SCALe]:RESet (No Query Form) 2363
[SENSe]:SGRam[:MEASview<y>]:WATerfall:Y[:SCALe] 2363
[SENSe]:SIGNalpath:PRIority 2364
[SENSe]:SIGNalvu:ACQuisition:CONTrol:HORizontal 2364
[SENSe]:SIGNalvu:ACQuisition:CONTrol:SAMPle:RATE 2365
[SENSe]:SIGNalvu:ACQuisition:CONTrol:SOURce<x>:STATe 2365
[SENSe]:SIGNalvu:ACQuisition:CONTrol:TRIGger:POSition 2366
[SENSe]:SIGNalvu:ACQuisition:CONTrol:VERTical 2366
[SENSe]:SIGNalvu:ACQuisition:DIGitizer:SAMPle:RATe 2367
[SENSe]:SIGNalvu:ACQuisition:DIGitizer:SAMPle:RATE:AUTo 2367
[SENSe]:SIGNalvu:ACQuisition:GLOBal:REFlevel 2368
[SENSe]:SIGNalvu:ACQuisition:INPuttype:REFlevel 2368
[SENSe]:SIGNalvu:ACQuisition:IQ:CALibrate (No query form) 2369
[:SENSe]:SIGNalvu:ACQuisition:NULL:IQ:OFFSet:CLEAR 2369
[:SENSe]:SIGNalvu:ACQuisition:PER:CHANnel<x>:COUPling 2369
[:SENSe]:SIGNalvu:ACQuisition:PER:CHANnel<x>:DESkew 2370
[:SENSe]:SIGNalvu:ACQuisition:PER:CHANnel<x>:EXTattn 2370

SENSe:SIGNalvu:ACQuisition:PER:CHANnel<x>:INSTrument: BWIDth (Query only) 2371
[:SENSe]:SIGNalvu:ACQuisition:PER:CHANnel<x>:REFlevel 2371
[SENSe]:SIGNalvu:ACQuisition:PER:CHANnel<x>:TERMination 2372
[SENSe]:SIGNalvu:ACQuisition:RESet (No query form) 2372
[:SENSe]:SIGNalvu:ACQuisition:SELEcted:CHANnel:TYPE 2373
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:CALIbrate:IQ 2373
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:CHANnel[:RF] 2373
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:CHANnel:I 2374
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:CHANnel:Q 2374
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:CHANnel:IPLus 2375
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:CHANnel:IMINus 2375
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:CHANnel:QPLus 2376
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:CHANnel:QMINus 2376
[:SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]: DOWNconverter:INJEctionside 2377
[:SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]: DOWNconverter:INSTrument:BANDwidth:MAXFreq 2377
[:SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]: DOWNconverter:INSTrument:BANDwidth:MINFreq 2378
[:SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]: DOWNconverter:LOFReq 2378
[:SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]: DOWNconverter:STATe 2379
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:PERSrc:REFlevel 2379
[SENSe]:SPECtrum[:MEASview<y>]:SELEcted:SOURce 2379
[SENSe]:SPECtrum[:MEASview<y>]:{BANDwidth |BWIDth}[:RESolution] 2380
[SENSe]:SPECtrum[:MEASview<y>]:{BANDwidth |BWIDth}[:RESolution]:ACTual? (Query Only) 2380
[SENSe]:SPECtrum[:MEASview<y>]:{BANDwidth|BWIDth} [:RESolution]:AUTO 2381
[SENSe]:SPECtrum[:MEASview<y>]:{BANDwidth|BWIDth} [:RESolution]:MODE 2381
[SENSe]:SPECtrum[:MEASview<y>]:{BANDwidth|BWIDth}:VIDeo 2382
[SENSe]:SPECtrum[:MEASview<y>]:{BANDwidth|BWIDth} :VIDeo:STATe 2383
[SENSe]:SPECtrum[:MEASview<y>]:CLEar:RESults (No Query Form) 2383
[SENSe]:SPECtrum[:MEASview<y>]:COLor 2383
[SENSe]:SPECtrum[:MEASview<y>]:COLor:MAXimum 2384
[SENSe]:SPECtrum[:MEASview<y>]:COLor:MINimum 2385
[SENSe]:SPECtrum[:MEASview<y>]:COLor:REVerse[:STATe] 2385
[SENSe]:SPECtrum[:MEASview<y>]:FFT:WINDow 2386
[SENSe]:SPECtrum[:MEASview<y>]:FILTer[:SHAPe] 2387
[SENSe]:SPECtrum[:MEASview<y>]:FREQuency:CENTer 2387
[SENSe]:SPECtrum[:MEASview<y>]:FREQuency:SPAN 2388
[SENSe]:SPECtrum[:MEASview<y>]:FREQuency:SPAN: BANDwidth[:RESolution]:RATio 2388
[SENSe]:SPECtrum[:MEASview<y>]:FREQuency:STARt 2389
[SENSe]:SPECtrum[:MEASview<y>]:FREQuency:STEP 2390
[SENSe]:SPECtrum[:MEASview<y>]:FREQuency:STEP:AUTO 2390
[SENSe]:SPECtrum[:MEASview<y>]:FREQuency:STOP 2391
[SENSe]:SPECtrum[:MEASview<y>]:LENGth 2391
[SENSe]:SPECtrum[:MEASview<y>]:LENGth:ACTual (Query Only) 2392
[SENSe]:SPECtrum[:MEASview<y>]:LENGth:AUTO 2392
[SENSe]:SPECtrum[:MEASview<y>]:MAX:SPAN (No Query Form) 2393
[SENSe]:SPECtrum[:MEASview<y>]:POINts:COUNt 2393
[SENSe]:SPECtrum[:MEASview<y>]:STARt 2394

[SENSe]:SPECtrum[:MEASview<y>]:STARt:AUTO 2394
[SENSe]:SPECtrum[:MEASview<y>]:TIMe:MODe 2395
[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth} 2395
[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth}:INTegration 2396
[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth}[:RESolution] 2396
[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth} [:RESolution]:AUTO 2397
[SENSe]:SPURious:CARRier:DETection 2397
[SENSe]:SPURious:CARRier:FREQuency 2398
[SENSe]:SPURious:CARRier:THReshold 2398
[SENSe]:SPURious:CLEar:RESults (No Query Form) 2399
[SENSe]:SPURious[:FREQuency]:OVERlap? (Query Only) 2399
[SENSe]:SPURious:LIST 2400
[SENSe]:SPURious:MODE 2400
[SENSe]:SPURious:OPTimization 2401
[SENSe]:SPURious:POINts:COUNt 2401
[SENSe]:SPURious:RANGe<x>:BANDwidth:VIDeo 2402
[SENSe]:SPURious:RANGe<x>:BANDwidth:VIDeo:STATe 2402
[SENSe]:SPURious:RANGe<x>:DETection 2403
[SENSe]:SPURious:RANGe<x>:EXCursion 2403
[SENSe]:SPURious:RANGe<x>:FILTer[:SHAPe] 2404
[SENSe]:SPURious:RANGe<x>:FILTer[:SHAPe]:BANDwidth 2404
[SENSe]:SPURious:RANGe<x>:FILTer[:SHAPe]:BANDwidth:AUTO 2405
[SENSe]:SPURious:RANGe<x>:FREQuency:STARt 2405
[SENSe]:SPURious:RANGe<x>:FREQuency:STOP 2406
[SENSe]:SPURious:RANGe<x>:LIMit:ABSolute:STARt 2406
[SENSe]:SPURious:RANGe<x>:LIMit:ABSolute:STOP 2407
[SENSe]:SPURious:RANGe<x>:LIMit:MASK 2407
[SENSe]:SPURious:RANGe<x>:LIMit:RELative:STARt 2408
[SENSe]:SPURious:RANGe<x>:LIMit:RELative:STOP 2409
[SENSe]:SPURious:RANGe<x>:STATe 2409
[SENSe]:SPURious:RANGe<x>:THReshold 2410
[SENSe]:SPURious:REFerence 2410
[SENSe]:SPURious:REFerence:MANual:POWer 2411
[SENSe]:SQUality[:MEASview<y>]:EVM:ALL[:STATe] 2411
[SENSe]:SQUality[:MEASview<y>]:SELEcted:SOURce 2411
[SENSe]:STABle[:MEASview<y>]:SELEcted:SOURce 2412
[SENSe]:SIGNalvu:ACQuisition:CONTrol:TRIGger 2412
[SENSe]:SIGNalvu:ACQuisition:CONTrol:CHANnel 2413
[SENSe]:SIGNalvu:ACQuisition:FFRame:ACTual? (Query only) 2413
[SENSe]:SIGNalvu:ACQuisition:FFRame:LIMit 2414
[SENSe]:SIGNalvu:ACQuisition:FFRame:STATe 2414
[SENSe]:TDIagram[:MEASview<y>]:SELEcted:SOURce 2415
SENSe:TOFF:POWEr:SPECtral:DENSity 2415
[SENSe]:TOVerview[:MEASview<y>]:FREQuency:CENTer 2416
[SENSe]:TOVerview[:MEASview<y>]:MAXTracepoints 2416
[SENSe]:TOVerview[:MEASview<y>]:TIME:PER:DIVision 2417

[SENSe]:TOVerview[:MEASview<y>]:SELEcted:SOURce 2417
[SENSe]:TRACe:SPURious:COUNt:RESet (No Query Form) 2418
[SENSe]:TRACe<x>:SPURious:COUNt 2418
[SENSe]:TRACe<x>:SPURious:COUNt:ENABle 2419
[SENSe]:TRACe<x>:SPURious:ENABle 2419
[SENSe]:TRACe<x>:SPURious:FREeze 2420
[SENSe]:TRACe<x>:SPURious:FUNCtion 2420
[SENSe]:TRACe<x>:SPURious:SELect 2421
SENSe:TXGain:{BANDwidth|BWIDth}[:RESolution] 2421
SENSe:TXGain:FREQuency:CENTer 2422
SENSe:TXGain:FREQuency:POINts 2422
SENSe:TXGain:FREQuency:SPAN 2423
SENSe:TXGain:FREQuency:STARt 2423
SENSe:TXGain:FREQuency:STEP 2424
SENSe:TXGain:FREQuency:STOP 2424
[SENSe]:USETtings (No Query Form) 2425
[SENSe]:WLAN:AD:ANALysis:BLOCkwise:CORRection:STATe 2425
[SENSe]:WLAN:AD:ANALysis:DC:OFFSet:CORRection:STATe 2426
[SENSe]:WLAN:AD:ANALysis:EQUAlizer:CONVergence 2426
[SENSe]:WLAN:AD:ANALysis:EQUAlizer:LENGth 2427
[SENSe]:WLAN:AD:ANALysis:EQUAlizer:MODE 2427
[SENSe]:WLAN:AD:ANALysis:EQUAlizer:RESet (No Query Form) 2428
[SENSe]:WLAN:AD:ANALysis:EQUAlizer:TAPS 2428
[SENSe]:WLAN:AD:ANALysis:EQUAlizer:TSRatio 2429
[SENSe]:WLAN:AD:ANALysis:EQUAlizer:TYPE 2429
[SENSe]:WLAN:AD:ANALysis:EQUAlizer:TYPE 2430
[SENSe]:WLAN:AD:ANALysis:HANDle:MULTi:PATH: CHANnel:STATe 2430
[SENSe]:WLAN:AD:ANALysis:LENGth 2431
[SENSe]:WLAN:AD:ANALysis:LENGth:ACTual? (Query Only) 2431
[SENSe]:WLAN:AD:ANALysis:LENGth:AUTO 2432
[SENSe]:WLAN:AD:ANALysis:OFFSet 2432
[SENSe]:WLAN:AD:ANALysis:OFFSet:AUTO 2433
[SENSe]:WLAN:AD:ANALysis:SWAP:IQ 2433
[SENSe]:WLAN:AD:ANALysis:TIMe:UNITs 2434
[SENSe]:WLAN:AD(:BANDwidth|:BWIDth):MODE 2434
[SENSe]:WLAN:AD(:BANDwidth|:BWIDth):TINTerval 2435
[SENSe]:WLAN:AD:EVM:BURSt:COUNt 2435
[SENSe]:WLAN:AD:EVM:BURSt:COUNt:ENABle 2436
[SENSe]:WLAN:AD:EVM:DATA:ANALyze:ALL:SYMBols 2436
[SENSe]:WLAN:AD:EVM:DATA:LENGth 2437
[SENSe]:WLAN:AD:EVM:DATA:STARt 2437
[SENSe]:WLAN:AD:EVM:UNITs 2438
[SENSe]:WLAN:AD:FILTer:ALPHa 2438
[SENSe]:WLAN:AD:FILTer:MEASurement 2439
[SENSe]:WLAN:AD:FREQuency:ERRor 2439
[SENSe]:WLAN:AD:FREQuency:ERRor:AUTO 2440

[SENSe]:WLAN:AD:MODUlation 2440
[SENSe]:WLAN:AD:SUMMary:CLEar:RESults (No Query Form) 2441
[SENSe]:WLAN:AD:SUMMary:LIMIt:EVM 2441
[SENSe]:WLAN:AD:SUMMary:LIMIt:FREQuency:ERROr 2442
[SENSe]:WLAN:ANALysis:LENGth 2442
[SENSe]:WLAN:ANALysis:LENGth:ACTual? (Query Only) 2443
[SENSe]:WLAN:ANALysis:LENGth:AUTO 2443
[SENSe]:WLAN:ANALysis:OFFSet 2443
[SENSe]:WLAN:AY:ANALysis:BLOCkwise:CORRection:STATe 2444
[SENSe]:WLAN:AY:ANALysis:EQUAlizer:CONVergence 2444
[SENSe]:WLAN:AY:ANALysis:DC:OFFSet:CORRection:STATe 2445
[SENSe]:WLAN:AY:ANALysis:EQUAlizer:LENGth 2446
[SENSe]:WLAN:AY:ANALysis:EQUAlizer:MODE 2446
[SENSe]:WLAN:AY:ANALysis:EQUAlizer:RESet (No Query Form) 2447
[SENSe]:WLAN:AY:ANALysis:EQUAlizer:TAPS 2447
[SENSe]:WLAN:AY:ANALysis:EQUAlizer:TSRatio 2448
[SENSe]:WLAN:AY:ANALysis:EQUAlizer:TYPE 2448
[SENSe]:WLAN:AY:ANALysis:LENGth 2449
[SENSe]:WLAN:AY:ANALysis:LENGth:ACTual? (Query Only) 2449
[SENSe]:WLAN:AY:ANALysis:LENGth:AUTO 2450
[SENSe]:WLAN:AY:ANALysis:OFFSet 2450
[SENSe]:WLAN:AY:ANALysis:OFFSet:AUTO 2451
[SENSe]:WLAN:AY:ANALysis:SFO:TRACk:STATe 2451
[SENSe]:WLAN:AY:ANALysis:SWAP:IQ 2452
[SENSe]:WLAN:AY:ANALysis:TIMe:UNITs 2453
[SENSe]:WLAN:AY:ANALysis:TRACk:EQ:STATe 2453
[SENSe]:WLAN:AY(:BANDwidth|:BWIDth):MODE 2454
[SENSe]:WLAN:AY(:BANDwidth|:BWIDth):TINTerval 2454
[SENSe]:WLAN:AY:CHbw 2455
[SENSe]:WLAN:AY:DUPBand 2455
[SENSe]:WLAN:AY:EVM:BURSt:COUNt 2456
[SENSe]:WLAN:AY:EVM:BURSt:COUNt:ENABle 2457
[SENSe]:WLAN:AY:EVM:DATA:ANALyze:ALL:SYMBols 2457
[SENSe]:WLAN:AY:EVM:DATA:LENGth 2458
[SENSe]:WLAN:AY:EVM:DATA:STARt 2458
[SENSe]:WLAN:AY:EVM:UNITs 2459
[SENSe]:WLAN:AY:FILTer:ALPHa 2459
[SENSe]:WLAN:AY:FILTer:MEASurement 2460
[SENSe]:WLAN:AY:FREQuency:ERRor 2460
[SENSe]:WLAN:AY:FREQuency:ERRor:AUTO 2461
[SENSe]:WLAN:AY:MODUlation 2461
[SENSe]:WLAN:AY:Region 2462
[SENSe]:WLAN:AY:SUMMary:CLEar:RESults (No Query Form) 2463
[SENSe]:WLAN:AY:SUMMary:LIMIt:EVM 2463
[SENSe]:WLAN:AY:SUMMary:LIMIt:FREQuency:ERROr 2464
[SENSe]:WLAN:BURSt 2464

[SENSe]:WLAN:CHANnel[:BANDwidth|:BWIDth] 2465
[SENSe]:WLAN:EQUalizer:TRAining 2465
[SENSe]:WLAN:FFT:LENGth? (Query Only) 2466
[SENSe]:WLAN:FFT:SRATe? (Query Only) 2466
[SENSe]:WLAN:FREQuency:ERRor 2467
[SENSe]:WLAN:FREQuency:ERRor:AUTO 2467
[SENSe]:WLAN:GUARd:INTerval 2468
[SENSe]:WLAN:MODulation:DETect:QAM1024 2468
[SENSe]:WLAN:MODulation:MANual 2469
[SENSe]:WLAN:MODulation:SOURce 2469
[SENSe]:WLAN:PILot:TRACking:AMPLitude:STATe 2470
[SENSe]:WLAN:PILot:TRACking:PHASe:STATe 2470
[SENSe]:WLAN:PILot:TRACking:TIMing:STATe 2471
[SENSe]:WLAN:RADix 2471
[SENSe]:WLAN:SCARriers 2471
[SENSe]:WLAN:SCARriers:RANGe:STARt 2472
[SENSe]:WLAN:SCARriers:RANGe:STOP 2472
[SENSe]:WLAN:SCARriers:SINGle:INDex 2473
[SENSe]:WLAN:SCARriers:SPACing 2473
[SENSe]:WLAN:SCARriers:SPACing:AUTO 2474
[SENSe]:WLAN:SIGNal:STANdard:TYPE 2474
[SENSe]:WLAN:SubCARrier:DERotation 2475
[SENSe]:WLAN:SUMMary:CLEar:RESults (No Query Form) 2475
[SENSe]:WLAN:SUMMary:EVM:RMS:AVERage:COUNt 2476
[SENSe]:WLAN:SUMMary:EVM:RMS:AVERage:COUNt:ENABle 2476
[SENSe]:WLAN:SWAP:IQ 2477
[SENSe]:WLAN:SYMBols|CHIPs 2477
[SENSe]:WLAN:SYMBols|CHIPs:MAX 2478
[SENSe]:WLAN:SYMBols|CHIPs:MAX:STATe 2478
[SENSe]:WLAN:SYMBols|CHIPs:RANGe:COUNt 2479
[SENSe]:WLAN:SYMBols|CHIPs:RANGe:STARt 2479
[SENSe]:WLAN:SYMBols|CHIPs:RANGe:STOP 2480
[SENSe]:WLAN:SYMBols|CHIPs:SINGle:INDex 2480
[SENSe]:WLAN:UNIT:FREQuency 2480
[SENSe]:WLAN:UNIT:TIME 2481
SOURce:TXGain:POWer 2481
*SRE 2482
Status commands 2483
STATus:ACPower:EVENts? (Query Only) 2483
STATus:{AM|FM|PM}:EVENts? (Query Only) 2483
STATus:AUDio:SPECtrum:EVENts? (Query Only) 2484
STATus:AUDio:SUMMary:EVENts? (Query Only) 2484
STATus:AVTime[:MEASview<y>]:EVENts? (Query Only) 2485
STATus:BIBEmissions:EVENts? (Query Only) 2486
STATus:BLUEtooth:CONSte:EVENts? (Query Only) 2486
STATus:BLUEtooth:EDIAgram:EVENts? (Query Only) 2487

STATus:BLUEtooth:FDVTime:EVENts? (Query Only) 2487
STATus:BLUEtooth:SUMMary:EVENts (Query Only) 2488
STATus:CCDF:EVENts? (Query Only) 2488
STATus:CONSte[:MEASview<y>]:EVENts? (Query only) 2489
STATus:DIQVtime[:MEASview<y>]:EVENts? (Query only) 2490
STATus:DPX:EVENts? (Query Only) 2490
STATus:EDIagram[:MEASview<y>]:EVENts? (Query only) 2491
STATus:EMC:EVENts? (Query Only) 2491
STATus:EVM[:MEASview<y>]:EVENts? (Query only) 2492
STATus:FDVTime[:MEASview<y>]:EVENts? (Query only) 2493
STATus:{FSETtling|PSETtling}:EVENts? (Query Only) 2493
STATus:FVTime[:MEASview<y>]:EVENts? (Query Only) 2494
STATus:IQVTime[:MEASview<y>]:EVENts? (Query Only) 2495
STATus:LTE:ACLR:EVENts? (Query Only) 2495
STATus:LTE:CHSPectrum:EVENts? (Query Only) 2496
STATus:LTE:CONSte:EVENts? (Query Only) 2496
STATus:LTE:PVTime:EVENts? (Query Only) 2497
STATus:MCPower:EVENts? (Query Only) 2497
STATus:MERRor[:MEASview<y>]:EVENts? (Query only) 2498
STATus:OBWidth:EVENts? (Query Only) 2498
STATus:OFDM:CONSte:EVENts? (Query Only) 2499
STATus:OFDM:EVM:EVENts? (Query Only) 2500
STATus:OFDM:MERRor:EVENts? (Query Only) 2500
STATus:OFDM:PERRor:EVENts? (Query Only) 2501
STATus:OFDM:POWer:EVENts? (Query Only) 2501
STATus:OFDM:STABle:EVENts? (Query Only) 2502
STATus:OPERation:CONDition? (Query Only) 2503
STATus:OPERation:ENABle 2503
STATus:OPERation[:EVENt]? (Query Only) 2504
STATus:OPERation:NTRansition 2504
STATus:OPERation:PTRansition 2505
STATus:P25:EDIagram:EVENts? (Query Only) 2505
STATus:P25:SUMMary:EVENts (Query Only) 2506
STATus:PERRor[:MEASview<y>]:EVENts? (Query only) 2506
STATus:PHVTime[:MEASview<y>]:EVENts? (Query Only) 2507
STATus:PNOise:EVENts? (Query Only) 2507
STATus:PRESet (No Query Form) 2508
STATus:PULSe:CUMulative:HISTogram[:MEASview<y>]:EVENts (Query Only) 2508
STATus:PULSe:CUMulative:STATistics[:MEASview<y>]:EVENts (Query Only) 2509
STATus:PULSe:OGRAm[:MEASview<y>]:EVENts (Query Only) 2510
STATus:PULSe:RESult:EVENts? (Query Only) 2510
STATus:PULSe:STATistics[:MEASview<y>]:EVENts? (Query Only) 2511
STATus:PULSe:TRACe[:MEASview<y>]:EVENts 2511
STATus:QUEStionable:CALibration:CONDition? (Query Only) 2512
STATus:QUEStionable:CALibration:ENABle 2512
STATus:QUEStionable:CALibration[:EVENt]? (Query Only) 2513

STATus:QUEStionable:CALibration:NTRansition 2513
STATus:QUEStionable:CALibration:PTRansition 2514
STATus:QUEStionable:CONDition? (Query Only) 2514
STATus:QUEStionable:ENABle 2515
STATus:QUEStionable[:EVENt]? (Query Only) 2515
STATus:QUEStionable:FREQuency:CONDition? (Query Only) 2516
STATus:QUEStionable:FREQuency:ENABle 2516
STATus:QUEStionable:FREQuency[:EVENt]? (Query Only) 2517
STATus:QUEStionable:FREQuency:NTRansition 2517
STATus:QUEStionable:FREQuency:PTRansition 2518
STATus:QUEStionable:NTRansition 2518
STATus:QUEStionable:PTRansition 2519
STATus:SEM:EVENts? (Query Only) 2519
STATus:SGRAM[:MEASview<y>]:EVENts (Query Only) 2520
STATus:SPECtrum[:MEASview<y>]:EVENts? (Query Only) 2521
STATus:SPURious:EVENts ? (Query Only) 2521
STATus:SQUality[:MEASview<y>]:EVENts? (Query only) 2522
STATus:TDIagram[:MEASview<y>]:EVENts? (Query only) 2522
STATus:TXGain:EVENts? (Query Only) 2523
STATus:WLAN:AD:CONSte:EVENts? (Query Only) 2523
STATus:WLAN:AD:EVM:EVENts? (Query Only) 2524
STATus:WLAN:AD:SUMMary:EVENts (Query Only) 2525
STATus:WLAN:AY:CONSte:EVENts? (Query Only) 2525
STATus:WLAN:AY:EVM:EVENts? (Query Only) 2526
STATus:WLAN:AY:SUMMary:EVENts? (Query Only) 2526
STATus:WLAN:CONSte:EVENts? (Query Only) 2527
STATus:WLAN:CRESponse:EVENts? (Query Only) 2527
STATus:WLAN:EVM:EVENts? (Query Only) 2528
STATus:WLAN:MERRor:EVENts? (Query Only) 2528
STATus:WLAN:PERRor:EVENts? (Query Only) 2529
STATus:WLAN:PVTime:EVENts? (Query Only) 2530
STATus:WLAN:STABle:EVENts? (Query Only) 2530
STATus:WLAN:SUMMary:EVENts? (Query Only) 2531
*STB? (Query Only) 2531
System commands 2532
SYSTem:ANTenna:AMPLifier? (Query Only) 2532
SYSTem:ANTenna:CAPabilities:REPort? (Query Only) 2532
SYSTem:ANTenna:CATalog? (Query Only) 2533
SYSTem:ANTenna:CONNect (No Query Form) 2533
SYSTem:ANTenna:DATA? (Query Only) 2533
SYSTem:ANTenna:DECLination? (Query Only) 2534
SYSTem:ANTenna:DECLination:USER 2534
SYSTem:ANTenna:DECLination:USER:STATe 2535
SYSTem:ANTenna:FREQuency:BAND? (Query Only) 2535
SYSTem:ANTenna:MANufacturer? (Query Only) 2536
SYSTem:ANTenna:MODel? (Query Only) 2536

SYSTem:ANTenna:OPTions? (Query Only) 2537
SYSTem:ANTenna:SELected 2537
SYSTem:ANTenna:SERial? (Query Only) 2538
SYSTem:ANTenna:SETDATA (No Query Form) 2538
SYSTem:BATTery:CCYCle? (Query Only) 2538
SYSTem:BATTery:DATE? (Query Only) 2539
SYSTem:BATTery:RCHarge? (Query Only) 2539
SYSTem:BATTery:SERial? (Query Only) 2540
SYSTem:BATTery:STATus? (Query Only) 2540
SYSTem:BATTery:TEMPerature? (Query Only) 2540
SYSTem:COMMunicate:GPIB[:SELF]:ADDRess 2541
SYSTem:DATE 2541
SYSTem:ERRor:ALL? (Query Only) 2542
SYSTem:ERRor:CODE:ALL? (Query Only) 2542
SYSTem:ERRor:CODE[:NEXT]? (Query Only) 2543
SYSTem:ERRor:COUNt? (Query Only) 2543
SYSTem:ERRor[:NEXT]? (Query Only) 2544
SYSTem:GNSS:ALTitude (No Query Form) 2544
SYSTem:GNSS:COURse? (Query Only) 2545
SYSTem:GNSS:DATA? (Query Only) 2545
SYSTem:GNSS:HDOP? (Query Only) 2546
SYSTem:GNSS:LATitude? (Query Only) 2546
SYSTem:GNSS:LONGitude? (Query Only) 2546
SYSTem:GNSS:SATellites? (Query Only) 2547
SYSTem:GNSS:SELected 2547
SYSTem:GNSS:SERial:BAUD 2548
SYSTem:GNSS:SERial:PORT 2548
SYSTem:GNSS:SPEed? (Query Only) 2548
SYSTem:GNSS:STATus? (Query Only) 2549
SYSTem:GNSS:TIMestamp? (Query Only) 2549
SYSTem:KLOCk 2550
SYSTem:LINK:SETTings 2550
SYSTem:LINK:SETTings:SYNC 2550
SYSTem:OPTions? (Query Only) 2551
SYSTem:PRESet (No Query Form) 2551
SYSTem:PRESet:APPLication:ACTion 2552
SYSTem:PRESet:BLUEtooth:STANdard 2552
SYSTem:PRESet:DPX (No Query Form) 2553
SYSTem:PRESet:DPX:ACTion 2553
SYSTem:PRESet:DPX:SELected 2553
SYSTem:PRESet:LTE:STANdard 2554
SYSTem:PRESet[:MAIN] (No Query Form) 2555
SYSTem:PRESet:MAIN:ACTion 2555
SYSTem:PRESet:MAIN:SELected 2555
SYSTem:PRESet:P25:STANdard (No Query Form) 2556
SYSTem:PRESet:STANdards (No Query Form) 2556

SYSTem:PRESet:STANdards:ACTion 2557
SYSTem:PRESet:USER (No Query Form) 2557
SYSTem:PRESet:USER:ACTion 2558
SYSTem:PRESet:USER:SELected 2558
SYSTem:PRESet:WLAN:AD:STANdard 2558
SYSTem:PRESet:WLAN:AY:STANdard 2559
SYSTem:PRESet:WLAN:STANdard 2560
SYSTem:SVPC:DISconnect (No Query Form) 2561
SYSTem:SVPC:CONNect (No Query Form) 2561
SYSTem:SVPC:CONNect:RSA300 (No Query Form) 2561
SYSTem:SVPC:INSTrument:SERial? (Query Only) 2561
SYSTem:SVPC:INSTrument:VERSion? (Query Only) 2562
SYSTem:SVPC:INSTrument:MODEL? (Query Only) 2562
SYSTem:SVPC:INSTrument:OPTions? (Query Only) 2562
SYSTem:SVPC:SOURce<x>:ADD 2563
SYSTem:SVPC:SOURce<x>:REMOve 2563
SYSTem:SVPC:SOURce<x>:REMOve:ALL 2564
SYSTem:SVPC:SOURce<x>:RENAme 2564
SYSTem:TIME 2565
SYSTem:VERSion? (Query Only) 2566
Trace Commands 2566
TRACe:{AM|FM|PM}:DETection 2566
TRACe:{AM|FM|PM}:FREeze 2567
TRACe:{AM|FM|PM}:FUNCtion 2567
TRACe:DIQVtime[:MEASview<y>]:ENABle:I 2567
TRACe:DIQVtime[:MEASview<y>]:ENABle:Q 2568
TRACe:DIQVtime[:MEASview<y>]:MODE 2568
TRACe:DIQVtime[:MEASview<y>]:SELect:I 2569
TRACe:DIQVtime[:MEASview<y>]:SELect:Q 2570
TRACe:EDIagram[:MEASview<y>]:ENABle:I 2570
TRACe:EDIagram[:MEASview<y>]:ENABle:Q 2571
TRACe:EDIagram[:MEASview<y>]:SELect:I 2571
TRACe:EDIagram[:MEASview<y>]:SELect:Q 2572
TRACe:EMC:AMBIent:COUNt 2572
TRACe:EMC:AMBIent:COUNt:ENABle 2573
TRACe:EMC:AMBIent:COUNt:RESet (No Query Form) 2573
TRACe:EMC:AMBIent:ENABle 2574
TRACe:EMC:AMBIent:FREeze 2574
TRACe:EMC:AMBIent:FUNCtion 2575
TRACe:EMC:AMBIent:SELect 2575
TRACe:EMC:MATH:ENABle 2576
TRACe:EMC:MATH:FREeze 2576
TRACe:EMC:MATH:LEFToperand 2577
TRACe:EMC:MATH:RIGHtoperand 2577
TRACe:EMC:MATH:SELect 2577
TRACe:EVM[:MEASview<y>]:MODE 2578

TRACe:FDVTime[:MEASview<y>]:MODE 2578
TRACe:{FSETtling|PSETtling}:AVERage:COUNt 2579
TRACe:{FSETtling|PSETtling}:AVERage:COUNt:CURRent? (Query Only) 2580
TRACe:{FSETtling|PSETtling}:AVERage:ENABle 2580
TRACe:{FSETtling|PSETtling}:RESet (No Query Form) 2580
TRACe:{FSETtling|PSETtling}:SMOothing:COUNt 2581
TRACe:{FSETtling|PSETtling}:SMOothing:ENABle 2581
TRACe<x>:FVTime[:MEASview<y>] 2582
TRACe<x>:FVTime[:MEASview<y>]:AVERage:COUNt 2582
TRACe<x>:FVTime[:MEASview<y>]:COLor:CURVe 2583
TRACe<x>:FVTime[:MEASview<y>]:COLor:SCALe:AUTO (No Query Form) 2583
TRACe<x>:FVTime[:MEASview<y>]:COUNt 2584
TRACe<x>:FVTime[:MEASview<y>]:COUNt:ENABle 2584
TRACe<x>:FVTime[:MEASview<y>]:COUNt:RESet (No Query Form) 2585
TRACe<x>:FVTime[:MEASview<y>]:DETection 2585
TRACe<x>:FVTime[:MEASview<y>]:DOT:PERSistent 2586
TRACe<x>:FVTime[:MEASview<y>]:DOT:PERSistent:TYPE 2586
TRACe<x>:FVTime[:MEASview<y>]:DOT:PERSistent:VARiable 2587
TRACe<x>:FVTime[:MEASview<y>]:FREeze 2587
TRACe<x>:FVTime[:MEASview<y>]:FUNCtion 2588
TRACe<x>:FVTime[:MEASview<y>]:SELect 2589
TRACe<x>:FVTime[:MEASview<y>]:SELect:LINe 2589
TRACe<x>:IQVTime[:MEASview<y>]:AVERage:COUNt 2590
TRACe<x>:IQVTime[:MEASview<y>]:COUNt 2590
TRACe<x>:IQVTime[:MEASview<y>]:COUNt:ENABle 2591
TRACe<x>:IQVTime[:MEASview<y>]:COUNt:RESet (No Query Form) 2591
TRACe<x>:IQVTime[:MEASview<y>]:DETection 2592
TRACe<x>:IQVTime[:MEASview<y>]:ENABle:I 2592
TRACe<x>:IQVTime[:MEASview<y>]:ENABle:Q 2593
TRACe<x>:IQVTime[:MEASview<y>]:FREeze 2593
TRACe<x>:IQVTime[:MEASview<y>]:FUNCtion 2594
TRACe<x>:IQVTime[:MEASview<y>]:SELect:I 2594
TRACe<x>:IQVTime[:MEASview<y>]:SELect:Q 2595
TRACe:MERRor[:MEASview<y>]:MODE 2596
TRACe:OBW:MAXHold 2596
TRACe:OFDM:CONSte 2597
TRACe:OFDM:CONSte:FREeze 2597
TRACe:OFDM:FLATness:FREeze 2597
TRACe:OFDM:FLATness:SHOW 2598
TRACe:PERRor[:MEASview<y>]:MODE 2598
TRACe<x>:PHVTime[:MEASview<y>] 2599
TRACe<x>:PHVTime[:MEASview<y>]:AVERage:COUNt 2600
TRACe<x>:PHVTime[:MEASview<y>]:COLor:CURVe 2600
TRACe<x>:PHVTime[:MEASview<y>]:COLor:SCALe:AUTO 2601
TRACe<x>:PHVTime[:MEASview<y>]:COUNt 2601
TRACe<x>:PHVTime[:MEASview<y>]:COUNt:ENABle 2602

TRACe<x>:PHVTime[:MEASview<y>]:COUNt:RESet (No Query Form) 2602
TRACe<x>:PHVTime[:MEASview<y>]:DETection 2603
TRACe<x>:PHVTime[:MEASview<y>]:DOT:PERSistent 2603
TRACe<x>:PHVTime[:MEASview<y>]:DOT:PERSistent:TYPE 2604
TRACe<x>:PHVTime[:MEASview<y>]:DOT:PERSistent:VARiable 2604
TRACe<x>:PHVTime[:MEASview<y>]:FREeze 2605
TRACe<x>:PHVTime[:MEASview<y>]:FUNCtion 2605
TRACe<x>:PHVTime[:MEASview<y>]:SELect 2606
TRACe<x>:PHVTime[:MEASview<y>]:SELect:LINe 2607
TRACe:SEM:COUNt 2607
TRACe:SEM:COUNt:RESet (No Query Form) 2608
TRACe:SEM:FUNCtion 2608
TRACe:SGRam[:MEASview<y>]:DETection 2609
TRACe:SGRam[:MEASview<y>]:FREeze 2609
TRACe:SGRam[:MEASview<y>]:FUNCtion 2610
TRACe:SGRam[:MEASview<y>]:FUNCtion:TIME 2611
TRACe:SGRam[:MEASview<y>]:RESet 2611
TRACe:SGRam[:MEASview<y>]:SELect:LINE 2612
TRACe:SPURious:COUNt 2612
TRACe:SPURious:COUNt:ENABle 2613
TRACe:SPURious:COUNt:RESet (No Query Form) 2613
TRACe:SPURious:FREeze 2613
TRACe:SPURious:FUNCtion 2614
TRACe:WLAN:AD:CONSte:TYPE 2614
TRACe:WLAN:AY:CONSte:TYPE 2615
TRACe:WLAN:FLATness:FREeze 2615
TRACe:WLAN:FLATness:SHOW 2616
TRACe1:TOVerview[:MEASview<y>] 2616
TRACe1:TOVerview[:MEASview<y>]:AVERage:COUNt 2617
TRACe1:TOVerview[:MEASview<y>]:COUNt 2617
TRACe1:TOVerview[:MEASview<y>]:COUNt:ENABle 2618
TRACe1:TOVerview[:MEASview<y>]:COUNt:RESet (No Query Form) 2618
TRACe1:TOVerview[:MEASview<y>]:DETection 2619
TRACe1:TOVerview[:MEASview<y>]:FREeze 2619
TRACe1:TOVerview[:MEASview<y>]:FUNCTion 2620
TRACe<x>:{AM|FM|PM} 2620
TRACe<x>:AVTime[:MEASview<y>] 2621
TRACe<x>:AVTime[:MEASview<y>]:AVERage:COUNt 2621
TRACe<x>:AVTime[:MEASview<y>]:AVERage:RESet (No Query Form) 2622
TRACe<x>:AVTime[:MEASview<y>]:COLor:CURVe 2622
TRACe<x>:AVTime[:MEASview<y>]:COLor:SCALe:AUTO (No Query Form) 2623
TRACe<x>:AVTime[:MEASview<y>]:COUNt 2623
TRACe<x>:AVTime[:MEASview<y>]:COUNt:ENABle 2624
TRACe<x>:AVTime[:MEASview<y>]:COUNt:RESet (No Query Form) 2624
TRACe<x>:AVTime[:MEASview<y>]:DETection 2625
TRACe<x>:AVTime[:MEASview<y>]:DOT:PERSistent 2626

TRACe<x>:AVTime[:MEASview<y>]:DOT:PERSistent:TYPE 2626
TRACe<x>:AVTime[:MEASview<y>]:DOT:PERSistent:VARiable 2627
TRACe<x>:AVTime[:MEASview<y>]:FREeze 2627
TRACe<x>:AVTime[:MEASview<y>]:FUNCtion 2628
TRACe<x>:AVTime[:MEASview<y>]:LEFToperand 2628
TRACe<x>:AVTime[:MEASview<y>]:RIGHtoperand 2629
TRACe<x>:AVTime[:MEASview<y>]:SELect 2630
TRACe<x>:AVTime[:MEASview<y>]:SELect:LINe 2630
TRACe<x>:BLUEtooth:CONSte 2631
TRACe<x>:BLUEtooth:CONSte:FREeze 2632
TRACe<x>:BLUEtooth:CONSte:MODE 2632
TRACe<x>:BLUEtooth:CONSte:SELect 2633
TRACe<x>:BLUEtooth:EDIagram:ENABle:I 2633
TRACe<x>:BLUEtooth:EDIagram:ENABle:Q 2633
TRACe<x>:BLUEtooth:EDIagram:SELect:I 2634
TRACe<x>:BLUEtooth:EDIagram:SELect:Q 2634
TRACe<x>:BLUEtooth:FDVTime:MODE 2635
TRACe<x>:CCDF:FREeze 2635
TRACe<x>:CCDF:SELect 2636
TRACe<x>:CCDF:SHOW 2636
TRACe<x>:CCDF:X 2637
TRACe<x>:CCDF:Y? (Query Only) 2637
TRACe<x>:CONSte[:MEASview<y>] 2638
TRACe<x>:CONSte[:MEASview<y>]:FREeze 2639
TRACe<x>:CONSte[:MEASview<y>]:MODE 2639
TRACe<x>:CONSte[:MEASview<y>]:Q:OFFSet 2640
TRACe<x>:CONSte[:MEASview<y>]:SELect (No query form) 2640
TRACe<x>:DPX 2641
TRACe<x>:DPX:AVERage:COUNt 2642
TRACe<x>:DPX:COLor:CURVe 2642
TRACe<x>:DPX:COLor:INTensity 2642
TRACe<x>:DPX:COLor:SCALe:AUTO (No Query Form) 2643
TRACe<x>:DPX:DETection 2643
TRACe<x>:DPX:DGRAM:SELect:LINE 2644
TRACe<x>:DPX:DOT:PERSistent 2644
TRACe<x>:DPX:DOT:PERSistent:TYPE 2645
TRACe<x>:DPX:DOT:PERSistent:VARiable 2645
TRACe<x>:DPX:FREeze 2646
TRACe<x>:DPX:FUNCtion 2646
TRACe<x>:DPX:LEFToperand 2647
TRACe<x>:DPX:RIGHtoperand 2647
TRACe<x>:DPX:SELect 2648
TRACe<x>:EMC:RECAll:ENABle 2648
TRACe<x>:EMC:RECAll:SELect 2649
TRACe<x>:EMC:SCAN:COUNt 2649
TRACe<x>:EMC:SCAN:COUNt:RESet (No Query Form) 2650

TRACe<x>:EMC:SCAN:ENABle 2650
TRACe<x>:EMC:SCAN:FUNCtion 2651
TRACe<x>:EMC:SCAN:SELect 2651
TRACe<x>:EMC:SPOT:ENABle 2652
TRACe<x>:EMC:SPOT:SELect 2652
TRACe<x>:{FSETtling|PSETtling}:FREeze 2653
TRACe<x>:{FSETtling|PSETtling}:SELect 2653
TRACe<x>:{FSETtling|PSETtling}:SHOW 2654
TRACe<x>:LTE:ACLR:COUNt 2654
TRACe<x>:LTE:ACLR:COUNt:RESet (No Query Form) 2655
TRACe<x>:LTE:ACLR:FUNCtion 2655
TRACe<x>:LTE:CONSte:SYNChronization:SEQUence:STATe? (Query Only) 2656
TRACe<x>:NOISe:FIGure:AVERage:COUNt 2656
TRACe<x>:NOISe:FIGure:COUNt:ENABle 2657
TRACe<x>:NOISe:FIGure:DETection 2657
TRACe<x>:NOISe:FIGure:FREeze 2658
TRACe<x>:NOISe:FIGure:FUNCtion 2658
TRACe<x>:NOISe:FIGure:SELect 2659
TRACe<x>:NOISe:FIGure:SHOW 2659
TRACe<x>:NOISe:GAIN:AVERage:COUNt 2660
TRACe<x>:NOISe:GAIN:COUNt:ENABle 2661
TRACe<x>:NOISe:GAIN:DETection 2661
TRACe<x>:NOISe:GAIN:FREeze 2662
TRACe<x>:NOISe:GAIN:FUNCtion 2662
TRACe<x>:NOISe:GAIN:SELect 2663
TRACe<x>:NOISe:GAIN:SHOW 2664
TRACe<x>:NOISe:TEMPerature:AVERage:COUNt 2664
TRACe<x>:NOISe:TEMPerature:COUNt:ENABle 2665
TRACe<x>:NOISe:TEMPerature:DETection 2665
TRACe<x>:NOISE:TEMPerature:FREeze 2666
TRACe<x>:NOISe:TEMPerature:FUNCtion 2666
TRACe<x>:NOISe:TEMPerature:SELect 2667
TRACe<x>:NOISe:TEMPerature:SHOW 2667
TRACe<x>:NOISe:YFACtor:AVERage:COUNt 2668
TRACe<x>:NOISe:YFACtor:COUNt:ENABle 2669
TRACe<x>:NOISe:YFACtor:DETection 2669
TRACe<x>:NOISE:YFACtor:FREeze 2670
TRACe<x>:NOISe:YFACtor:FUNCtion 2670
TRACe<x>:NOISe:YFACtor:SELect 2671
TRACe<x>:NOISe:YFACtor:SHOW 2671
TRACe<x>:OFDM:EVM 2672
TRACe<x>:OFDM:EVM:FREeze 2672
TRACe<x>:OFDM:EVM:SELect 2673
TRACe<x>:OFDM:MERRor 2673
TRACe<x>:OFDM:MERRor:FREeze 2674
TRACe<x>:OFDM:MERRor:SELect 2674

TRACe<x>:OFDM:PERRor 2675
TRACe<x>:OFDM:PERRor:FREeze 2675
TRACe<x>:OFDM:PERRor:SELect 2676
TRACe<x>:OFDM:POWer 2676
TRACe<x>:OFDM:POWer:FREeze 2677
TRACe<x>:OFDM:POWer:SELect 2677
TRACe<x>:P25:CONStellation 2678
TRACe<x>:P25:CONStellation:FREeze 2678
TRACe<x>:P25:CONStellation:MODE 2679
TRACe<x>:P25:CONStellation:SELect 2679
TRACe<x>:P25:EDIagram:ENABle:I 2680
TRACe<x>:P25:EDIagram:ENABle:Q 2680
TRACe<x>:P25:EDIagram:SELect:I 2681
TRACe<x>:P25:EDIagram:SELect:Q 2682
TRACe<x>:PNOise:SELect 2682
TRACe<x>:PNOise:SHOW 2683
TRACe<x>:PNOise:SMOothing:COUNt 2683
TRACe<x>:PNOise:SMOothing:ENABle 2683
TRACe<x>:PNOise:SMOothing:RESet (No Query Form) 2684
TRACe<x>:SPECtrum[:MEASview<y>] 2684
TRACe<x>:SPECtrum[:MEASview<y>]:AVERage:COUNt 2685
TRACe<x>:SPECtrum[:MEASview<y>]:AVERage:RESet (No Query Form) 2685
TRACe<x>:SPECtrum[:MEASview<y>]:COLor:CURVe 2686
TRACe<x>:SPECtrum[:MEASview<y>]:COLor:SCALe:AUTO (No Query Form) 2687
TRACe<x>:SPECtrum[:MEASview<y>]:COUNt 2687
TRACe<x>:SPECtrum[:MEASview<y>]:COUNt:ENABle 2688
TRACe<x>:SPECtrum[:MEASview<y>]:COUNt:RESet (No Query Form) 2688
TRACe<x>:SPECtrum[:MEASview<y>]:DETection 2689
TRACe<x>:SPECtrum[:MEASview<y>]:DOT:PERSistent 2689
TRACe<x>:SPECtrum[:MEASview<y>]:DOT:PERSistent:TYPE 2690
TRACe<x>:SPECtrum[:MEASview<y>]:DOT:PERSistent:VARiable 2690
TRACe<x>:SPECtrum[:MEASview<y>]:FREeze 2691
TRACe<x>:SPECtrum[:MEASview<y>]:FUNCtion 2691
TRACe<x>:SPECtrum[:MEASview<y>]:LEFToperand 2692
TRACe<x>:SPECtrum[:MEASview<y>]:RIGHtoperand 2693
TRACe<x>:SPECtrum[:MEASview<y>]:SELect 2693
TRACe<x>:TXGain 2694
TRACe<x>:TXGain:AVERage:COUNt 2694
TRACe<x>:TXGain:AVERage:RESet (No Query Form) 2695
TRACe<x>:TXGain:COUNt 2695
TRACe<x>:TXGain:COUNt:ENABle 2696
TRACe<x>:TXGain:COUNt:RESet (No Query Form) 2696
TRACe<x>:TXGain:FREeze 2697
TRACe<x>:TXGain:FUNCtion 2697
TRACe<x>:TXGain:SELect 2698
TRACe<x>:WLAN:AD:CONSte:RESUlt:CONTent:STATe 2698

TRACe<x>:WLAN:AY:CONSte:RESUlt:CONTent:STATe 2699
TRACe<x>:WLAN:CONSte 2700
TRACe<x>:WLAN:CONSte:FREeze 2700
TRACe<x>:WLAN:CRESponse:SELect 2701
TRACe<x>:WLAN:EVM:FREeze 2701
TRACe<x>:WLAN:EVM[:SHOW] 2702
TRACe<x>:WLAN:MERRor:FREeze 2702
TRACe<x>:WLAN:MERRor[:SHOW] 2703
TRACe<x>:WLAN:PERRor:FREeze 2703
TRACe<x>:WLAN:PERRor[:SHOW] 2704
Trigger commands 2704
*TRG (No Query Form) 2704
TRIGger:ADVanced:SWEep:MODE 2705
TRIGger:DPX:SHOW:FRAMes 2705
TRIGger:DPX:TDM:SHOW:ALL 2706
TRIGger:EVENt:EXTernal:SLOPe 2706
TRIGger:EVENt:INPut:LEVel 2707
TRIGger:EVENt:INPut:SLOPe 2707
TRIGger:EVENt:INPut:TYPE 2707
TRIGger:EVENt:SOURce 2708
TRIGger:MASK:NEW (No Query Form) 2708
TRIGger:MASK:NEW:AUTO (No Query Form) 2709
TRIGger:MASK:OPEN (No Query Form) 2710
TRIGger:MASK:SAVE (No Query Form) 2711
TRIGger:SAVE:COUNt:MAXimum 2711
TRIGger:SAVE:COUNt[:STATe] 2711
TRIGger:SAVE:DATA:FORMat 2712
TRIGger:SAVE:DATA[:STATe] 2712
TRIGger:SAVE:PICTure:FORMat 2713
TRIGger:SAVE:PICTure[:STATe] 2713
TRIGger[:SEQuence]:ADVanced:HOLDoff 2713
TRIGger[:SEQuence]:ADVanced:HOLDoff:ENABle 2714
TRIGger[:SEQuence]:ADVanced:SWEep:MODE 2714
TRIGger[:SEQuence]:ADVanced:SWEPt:SEGMent:ENABle 2715
TRIGger[:SEQuence]:EVENt:A:EDGE:COUPling 2715
TRIGger[:SEQuence]:EVENt:A:EDGE:HOLDoff:MODE 2716
TRIGger[:SEQuence]:EVENt:A:EDGE:HOLDoff:TIME 2716
TRIGger[:SEQuence]:EVENt:A:EDGE:LEVEl 2716
TRIGger[:SEQuence]:EVENt:EXTFront:IMPedance 2717
TRIGger[:SEQuence]:EVENt:EXTFront:LEVel 2717
TRIGger[:SEQuence]:EVENt:EXTFront:SLOPe 2718
TRIGger[:SEQuence]:EVENt:EXTRear:SLOPe 2718
TRIGger[:SEQuence]:EVENt:GATed 2718
TRIGger[:SEQuence]:EVENt:INPut:DDENsity:AMPLitude 2719
TRIGger[:SEQuence]:EVENt:INPut:DDENsity:AMPLitude:TOLerance 2719
TRIGger[:SEQuence]:EVENt:INPut:DDENsity:FREQuency 2720

TRIGger[:SEQuence]:EVENt:INPut:DDENsity:FREQuency:TOLerance 2720
TRIGger[:SEQuence]:EVENt:INPut:DDENsity:THReshold 2721
TRIGger[:SEQuence]:EVENt:INPut:DDENsity:VIOLation 2721
TRIGger[:SEQuence]:EVENt:INPut:FEDGe:LEVel 2722
TRIGger[:SEQuence]:EVENt:INPut:FEDGe:LEVel:ACTual? (Query Only) 2722
TRIGger[:SEQuence]:EVENt:INPut:FEDGe:SLOPe 2723
TRIGger[:SEQuence]:EVENt:INPut:FEDGe:THReshold:LEVel 2723
TRIGger[:SEQuence]:EVENt:INPut:FEDGe:THReshold:STATe 2724
TRIGger[:SEQuence]:EVENt:INPut:FMASk:{BANDwidth|BWIDth}[:RESolution] 2724
TRIGger[:SEQuence]:EVENt:INPut:FMASk:BANDwidth|BWIDth[:RESolution]:ACTual? (Query Only) 2725
TRIGger[:SEQuence]:EVENt:INPut:FMASk:BANDwidth|BWIDth[:RESolution]:AUTO 2725
TRIGger[:SEQuence]:EVENt:INPut:FMASk:VIOLation 2725
TRIGger[:SEQuence]:EVENt:INPut:LEVel 2726
TRIGger[:SEQuence]:EVENt:INPut:RUNT:PULSe 2727
TRIGger[:SEQuence]:EVENt:INPut:RUNT:PULse:HIGH:LEVel 2727
TRIGger[:SEQuence]:EVENt:INPut:RUNT:PULSe:LOW:LEVel 2728
TRIGger[:SEQuence]:EVENt:INPut:SLOPe 2728
TRIGger[:SEQuence]:EVENt:INPut:TDBWidth 2728
TRIGger[:SEQuence]:EVENt:INPut:TDBWidth:ACTual? (Query Only) 2729
TRIGger[:SEQuence]:EVENt:INPut:TDBWidth:STATe 2729
TRIGger[:SEQuence]:EVENt:INPut:TYPE 2730
TRIGger[:SEQuence]:EVENt:SOURce 2730
TRIGger[:SEQuence]:FORCed 2731
TRIGger:SEQuence:FORCed (No Query Form) 2731
TRIGger[:SEQuence]:IMMediate (No Query Form) 2732
TRIGger[:SEQuence]:STATus 2732
TRIGger[:SEQuence][:STATus] 2733
TRIGger[:SEQuence]:TIME:DELay 2733
TRIGger[:SEQuence]:TIME:POSition 2733
TRIGger[:Sequence]:TIME:POSition:AUTO 2734
TRIGger[:SEQuence]:TIME:QUALified 2734
TRIGger[:SEQuence]:TIME:QUALified:TIME<x> 2735
TRIGger[:SEQuence]:EVENt:A:EDGE:SETTings 2735
TRIGger[:SEQuence]:EVENt:A:EDGE:SLOPe 2736
TRIGger[:SEQuence]:EVENt:A:EDGE:SOURce 2736
TRIGger[:SEQuence]:EVENt:A:TYPE 2737
TRIGger[:SEQuence]:MODE 2737
TRIGger:TIME:POSition 2737
TRIGger:TIME:POSition:AUTO 2738
UNIT:POWer 2738
*WAI (No Query Form) 2739

Status and Events 2740
Status and event reporting system 2740
Status Byte 2742
Standard event status block 2744

Operation status block . . . 2744
Questionable Status Block . . . 2745
Queues . . . 2747
Status and event processing sequence . . . 2747
Synchronizing execution . . . 2748
Error messages and codes . . . 2750
Command errors . . . 2750
Execution errors . . . 2751
Device specific errors . . . 2753
Query Errors . . . 2754
Device errors . . . 2754
Status Conditions . . . 2758
Global status . . . 2758
Measurement status . . . 2758
Source status . . . 2760

Appendices . . . 2762
Appendix A: Character charts . . . 2762
Appendix B: SCPI conformance information . . . 2763
Appendix C: Creating a DPX Frame Access Plugin . . . 2764

Glossary . . . 2767

Preface

This document provides supplementary information about the remote commands for the SignalVu Vector Signal Analysis Software. The SignalVu software enables you to use the MSO/DPO5000, DPO7000, and DPO/DSA/MSO70000 Series Oscilloscopes to analyze RF signal behavior with the same software used on the RSA6000 Series Real-Time Analyzers. The remote command details are described in the following manuals that can be downloaded from www.tektronix.com/manuals.

- RSA6100B Series and RSA5100A Series Real-Time Spectrum Analyzers Programmer Manual
- MSO/DPO5000, DPO7000/C, DPO70000/B/C, DSA70000/B/C, and MSO70000/C Series Digital Oscilloscopes Programmer Manual

The SignalVu software supports the use of the RSA6100B Series commands with some limitations. For example, commands that control output functions are not supported. The details are given in the Measurement Display Support and Remote Command Support sections of this manual. Some commands are added for the SignalVu software and are described in the Additional Commands section. The error messages and codes are listed in the Error Messages and Codes section.

This manual is composed of the following sections:

- *Getting Started* outlines how to use the GPIB interface.
- *Syntax and Commands* defines the syntax used in command descriptions, presents a list of all command subsystems, and presents detailed descriptions of all programming commands.
- *Status and Events* describes how the status and Events Reporting system operates and presents a list of all system errors.
- *Appendices* provides additional information including character charts, GPIB interface specification, and factory initialization settings.

Related documentation

- SignalVu Vector Signal Analysis Software Help

  The help contains detailed information about how to operate the application and is found in the SignalVu application. A PDF of the Help is also available at tek.com/manuals.
- *TekVISA Programmer Manual*

  (Tektronix part number 077-0140-XX)

  This manual is available as a printable PDF file on the Tektronix Web site (tek.com/manuals). The manual describes TekVISA, the Tektronix implementation of the VISA Application Programming Interface (API). TekVISA is industry-compliant software for writing interoperable instrument drivers in a variety of Application Development Environments (ADEs).

Getting Started

You can write computer programs that remotely take measurements and read those measurements for further analysis or storage. To help you get started with programming, this section includes the following subsections.

- Overview of the Manual

  Summarizes each major section of this manual.
- Measurement Support

  Describes which RSA measurement displays are available with the software.
- Setting Up Remote Communications

  Describes how to set up an oscilloscope to a controller and set a unique device address.
- Remote Command Support

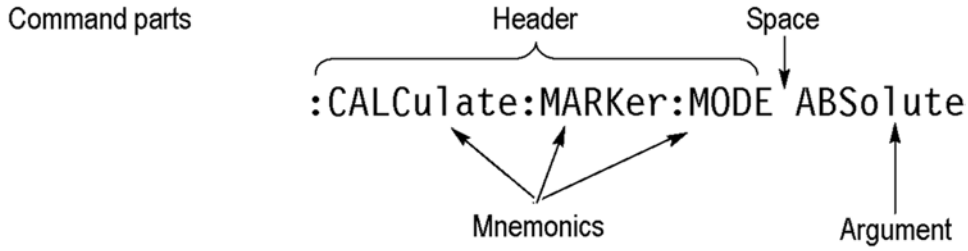
  Describes which groups of RSA commands are supported or not supported by the software.

Overview of the Manual

The information contained in each major section of this manual is described below.

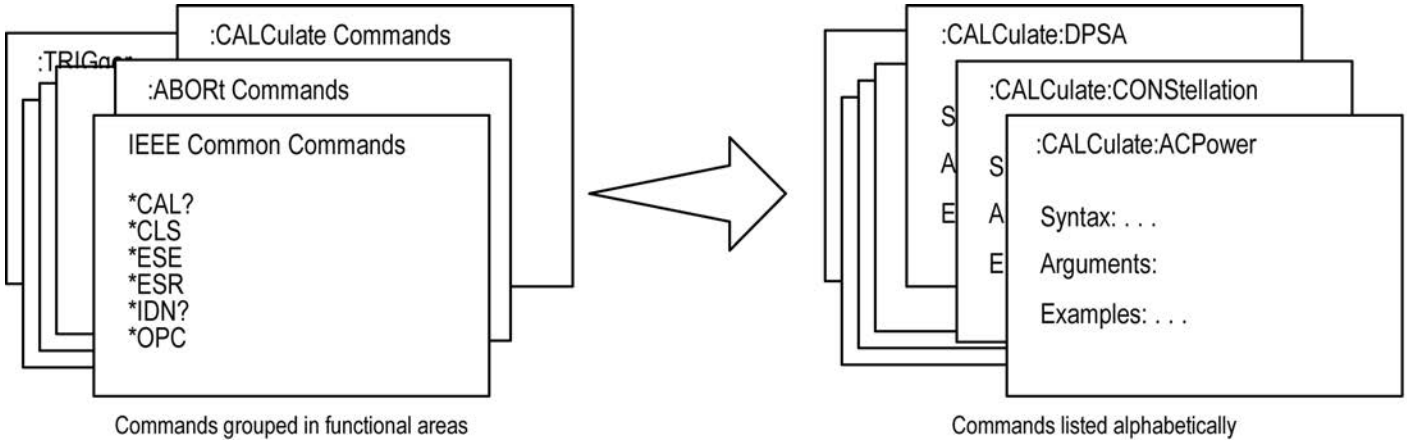
Syntax and Commands

Syntax and Commands (on page 118) describes the structure and content of the messages your program sends to the analyzer. The following figure shows command parts as described in Command Syntax (on page 118).



Command parts

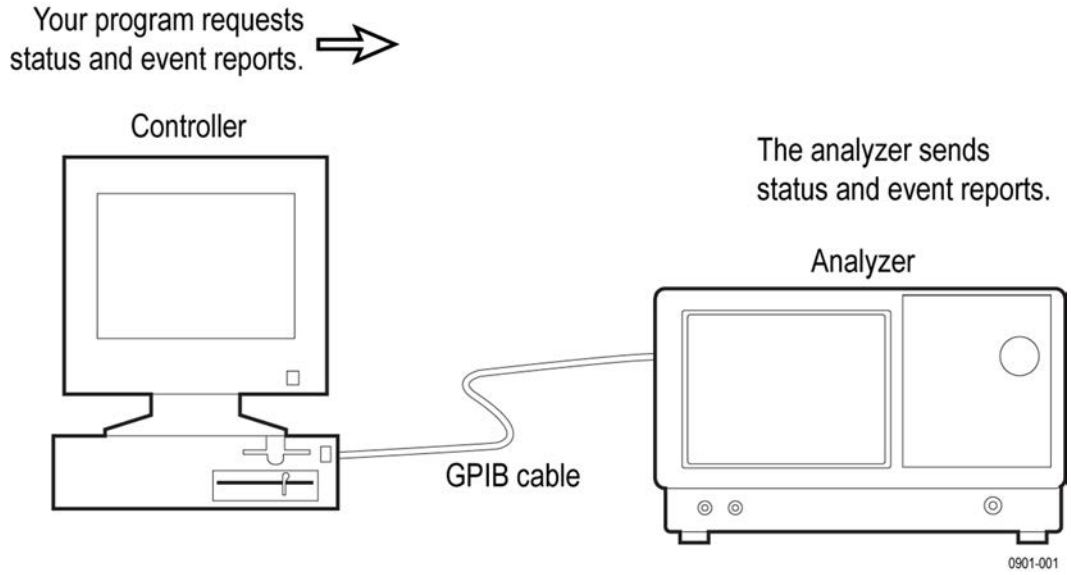
Command Descriptions (on page 342) also describes the effect of each command and provides examples of how you might use it. The Command Groups (on page 124) subsection provides lists by functional areas. The commands are listed alphabetically in the Command Descriptions (on page 342) section.



Functional groupings and an alphabetical list of commands

Status and Events

The program may request information from the instrument. The instrument provides information in the form of status and error messages. The following figure illustrates the basic operation of this system. Status and Events (on page 2740) describes how to get status or event information from the program and details the event and error messages.



Event-driven program

Setting up remote communications

Connect your oscilloscope to a controller and set a unique device address. The default settings for the GPIB configuration are:

- GPIB Address 1
- GPIB ModeGPIB Talk/Listen

To change either of the GPIB settings, do the following:

1. Select **GPIB Configuration** from the Utilities menu.
2. Click the **Configuration Talk/Listen** button.
3. Change the **GPIB Address** to a unique address.
4. Click the Close button.

The instrument is now set up for bidirectional communication with your controller.For details, refer to the Programmer manual for your oscilloscope.

Measurement support

The RSA6100B Series analyzers support various measurements, which are categorized into measurement view groups. The following table lists all measurement displays available on the RSA6100B Series analyzers. The columns at right show whether or not a measurement is supported by the SignalVu software. Required options, if applicable, are noted in the left column.

- If you send a command for the measurement view that is not active on screen, an execution error will occur.
- Some measurement displays require installed options. See the Tektronix Web site product page for a list of available options.

Measurement support

Measurement group	Measurement
General signal viewing	Amplitude vs Time
	Frequency vs Time
	Phase vs Time
	RF I&Q vs Time
	Spectrogram
	Spectrum
	Time overview
Analog Modulation	Amplitude modulation
	Frequency modulation
	Phase modulation
Phase Noise and Jitter	Phase Noise Display
Settling Time (requires Option SV12)	Freq Settling
	Phase Settling
General purpose digital modulation (requires Option SV21)	Constellation
	Demod I&Q versus Time
	EVM vs Time
	Eye Diagram
	Freq Dev vs Time
	Mag Error vs Time
	Phase Error vs Time
	Signal quality
	Symbol table
	Trellis
	User defined filters

Measurement group	Measurement
RF	Channel power and ACPR
	Freq Settling (requires Option SVT)
	MCPR
	Occupied Bandwidth
	Phase Settling (requires Option SVT)
	Signal Strength (requires Option MAP)
	SEM
	Spurious
OFDM (requires Option SVO)	OFDM Channel Response
	OFDM Constellation
	OFDM EVM
	OFDM Mag Error
	OFDM Phase Error
	OFDM Power
	OFDM Summary
	OFDM Symbol Table
	OFDM Flatness

Measurement group	Measurement
WLAN (requires Option SV23,SV24, or SV25)	WLAN Channel Response
	WLAN Constellation
	WLAN EVM
	WLAN Flatness
	WLAN Mag Error
	WLAN Phase Error
	WLAN Power vs Time
	WLAN Summary
	WLAN Symbol Table
	SEM
P25 Analysis (requires Option SV26)	P25 Eye Diagram
	P25 Freq Dev vs Time
	P25 Power vs Time
	P25 Constellation
	P25 Symbol Table
	P25 Summary
	MCPR
	Time Overveiw

Measurement group	Measurement
Bluetooth Analysis (requires Option SV27)	BT 20dB BW
	BT CF Offset and Drift
	BT Constellation
	BT Eye Diagram
	BT Freq Dev vs Time
	BT Inband Emission
	BT Summary
	BT Symbol Table
	Time Overview
LTE Analysis (requires Option SV28)	LTE ACLR
	LTE Channel Spectrum
	LTE Constellation
	LTE Power vs Time
	SEM
Pulsed RF (requires Option SVP)	Pulse statistics
	Pulse table
	Pulse trace
802.11ad Analysis (requires Option SV30)	802.11ad EVM vs Time
	802.11ad Symbol Table
	802.11ad Constellation
	802.11ad EVM vs Time
Audio Analysis(requires Option SVA)	Audio Spectrum
	Audio Summary

Remote command support

The SignalVu software supports the use of the RSA6100B Series analyzer remote commands with some limitations. The following table summarizes the RSA6100B Series commands supported and not supported by the SignalVu software for each command group. For details on the RSA6100B Series remote commands, refer to the RSA6100B Programmer Manual. It is available for download at www.tektronix.com/downloads.

Command support summary

Command group	Function	Commands supported	Commands not supported
IEEE common	Conforms to the IEEE Std 488.2.	Use the oscilloscope IEEE	common commands.[1]
CALCulate	Controls the markers and the search operations.	All	CALCulate:DPSA:
CALibration	Calibrates the instrument.	Use the oscilloscope	CALibrate commands[1]
DISPlay	Controls the presentation of text, graph, and trace.	All	DISPlay:DPSA: DISPlay:CCDF:
FETCh	Retrieves the measurements from the last acquired data.	All except in the right box.	FETCh:RFIN:IQ:SCALe? FETCh:DPSA:
INITiate	Controls the initiation of data acquisition.	All	
INPut	Controls the characteristics of the signal input.	All except in the right box.	INPut[:RF]:ATTenuation INPut[:RF]:ATTenuation:AUTO INPut[:RF]:ATTenuation:MONitor:STATe INPut[:RF]:GAIN:STATe
MMEMory	Provides mass storage capabilities for the instrument. Setup and TIQ files (:MMEMory:xxx:IQ and :MMEMory:xx:STATe) contain complete oscilloscope setup information. When one of these files is loaded, the oscilloscope is setup according to the saved oscilloscope settings in the file.	All except in the right box. (See *Note* following this table.) You can also use the FILESystem commands of the oscilloscope.[1]	MMEMory:CALibration:LOAD:CORRection :EXTernal:EDIT<x> MMEMory:CALibration:STORe:CORRection :EXTernal:EDIT<x> MMEMory:DPSA:
OUTPut	Controls the characteristics of the signal output.	-	All
READ	Obtains the measurement results by acquiring fresh data.	All except in the right box.	READ:DPSA

Command group	Function	Commands supported	Commands not supported
SENSe	Sets up detailed conditions for each measurement.	All except in the right box. (See *Note* following this table.) You can also use the additional commands.	[SENSe]:ACQuisition:FFRame:ACTual? [SENSe]:ACQuisition:FFRame:Limit [SENSe]:ACQuisition:FFRame:STATe [SENSe]:ANALysis:ADVanced:DITHer [SENSe]:ROSCillator:SOURce [SENSe]:DPSA:
STATus	Controls the status and event registers.	-	All
SYSTem	Sets or queries system parameters for operation.	All except in the right box.	SYSTem:COMMunicate:GPIB[:SELF]:ADDRess SYSTem:DATE SYSTem:TIME SYSTem:VERSion SYSTem:HW:VERSion
TRACe	Controls trace activation and math operations.	All except in the right box.	TRACe:DPSA
TRIGger	Controls triggering.	Use the oscilloscope	TRIGger commands.[1]
UNIT	Specifies fundamental units for measurement.	-	- UNIT:POWer (Use the [SENSe]:POWer:UNITs command.)

[1] Refer to the *MSO/DPO5000, DPO7000/C, DPO70000/B/C, DSA70000/B/C, and MSO70000/C Series Digital Oscilloscopes Programmer Manual.*.

Quick Tips

- The :MMEMory:LOAD:IQ command accepts the file extensions other than .tiq. In the RSA6100B Series spectrum analyzers, this command ignores the file extension passed in (if any), and assumes that it is .tiq. In the SignalVu software, the file extension can be either .tiq or .wfm, and so is not ignored. If no file extension is included, it is assumed to be .tiq.
- To avoid conflicts with the oscilloscope commands, use the following syntax.
  - Use the command header [SENSe]:SIGNalvu:ACQuisition for the [SENSe]:ACQuisition commands. For example, replace the [SENSe]:ACQuisition:BANDwidth command with [SENSe]:SIGNalvu:ACQuisition:BANDwidth.
  - Replace the [SENSe]:MEASurement:FREQuency command with [SENSe]:FREQuency.

Syntax and Commands

Command syntax

This section contains information on the Standard Commands for Programmable Instruments (SCPI) and IEEE 488.2 Common Commands you can use to program your RSA Series analyzer. The information is organized in the following subsections

- Backus-Naur Form Definition
- SCPI Commands and Queries
- IEEE 488.2 Common Commands
- Constructed Mnemonics

Backus-Naur Form Definition

This manual may describe commands and queries using the Backus-Naur Form (BNF) notation. The following table defines the standard BNF symbols.

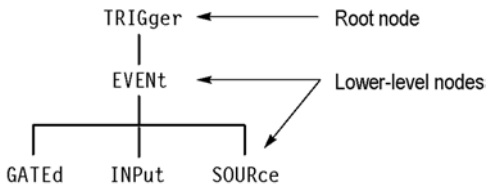
BNF symbols and meanings

Symbol	Meaning
< >	Defined element
:=	Is defined as
|	Exclusive OR
{ }	Group; one element is required
[ ]	Optional; can be omitted
. . .	Previous element(s) may be repeated
( )	Comment

SCPI Commands and Queries

SCPI is a standard created by a consortium that provides guidelines for remote programming of instruments. These guidelines provide a consistent programming environment for instrument control and data transfer. This environment uses defined programming messages, instrument responses, and data format across all SCPI instruments, regardless of manufacturer. The analyzer uses a command language based on the SCPI standard.

The SCPI language is based on a hierarchical or tree structure as shown in the following figure that represents a subsystem. The top level of the tree is the root node; it is followed by one or more lower-level nodes.



Example of SCPI subsystem hierarchy tree

You can create commands and queries from these subsystem hierarchy trees.Commands specify actions for the instrument to perform. Queries return measurement data and information about parameter settings.

Creating Commands

SCPI commands are created by stringing together the nodes of a subsystem hierarchy and separating each node by a colon.

In the figure above, TRIGger is the root node and EVENt, GATed, INPut, and SOURce are lower-level nodes. To create a SCPI command, start with the root node TRIGger and move down the tree structure adding nodes until you reach the end of a branch. Most commands and some queries have parameters; you must include a value for these parameters. If you specify a parameter value that is out of range, the parameter will be set to a default value. The command descriptions,list the valid values for all parameters.

For example, TRIGgerEVENt:SOURce EXTRear is a valid SCPI command created from the hierarchy tree. (See Syntax and Commands (on page 110).)

Creating Queries

To create a query, start at the root node of a tree structure, move down to the end of a branch, and add a question mark. TRIGgerEVENt:SOURce? is an example of a valid SCPI query using the hierarchy tree in the figure. (See Syntax and Commands (on page 110).)

Query Responses

The query causes the analyzer to return information about its status or settings.When a query is sent to the analyzer, only the values are returned. When the returned value is a mnemonic, it is noted in abbreviated format, as shown in the following table.

Query response examples

Query	Response
CALCulate:SPECtrum:MARKer:X?	7.50E+9
SENSe:SPECtrum:FFT:WINDow?	BH4B

A few queries also initiate an operation action before returning information. For example, the *CAL? query runs a calibration.

Parameter Types

Every parameter in the command and query descriptions is of a specified type.The parameters are enclosed in brackets, such as <value>. The parameter type is listed after the parameter and is enclosed in parentheses, for example, (Boolean).Some parameter types are defined specifically for the RSA Series command set and some are defined by ANSI/IEEE 488.2-1987 as shown in the following table.

Parameter types used in syntax descriptions

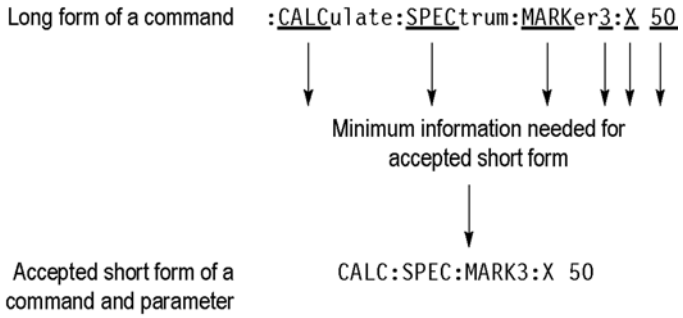
Parameter type	Description	Example
arbitrary block	A specified length of arbitrary data; defined in ANSI/IEEE 488.2 as "Definite Length Arbitrary Block Response Data"	#512234xxxxx . . . where 5 indicates that the following 5 digits (12234) specify the length of the data in bytes;xxxxx ... indicates the data
Boolean	Boolean numbers or values	ON or 1; OFF or 0
binary	Binary numbers	#B0110
octal	Octal numbers	#Q57, #Q3
hexadecimal	Hexadecimal numbers (0-9, A, B, C, D, E, F); an ANSI/IEEE 488.2-1992-defined parameter type	#HAA, #H1
NR1 numeric	Integers; an ANSI/IEEE 488.2-1992-defined parameter type	0, 1, 15, -1
NR2 numeric	Decimal numbers; an ANSI/IEEE 488.2-1992-defined parameter type; some commands and queries will accept an octal or hexadecimal value even though the parameter type is defined as NR2	1.2, 3.141516, -6.5
NR3	Floating point numbers; an ANSI/IEEE 488.2-1992-defined parameter type.	3.1415E-9, -16.1E5
NRf numeric	Flexible decimal number that may be type NR1, NR2 or NR3; anANSI/IEEE 488.2-1992-defined parameter type.	See NR1, NR2, and NR3 examples
string	Alphanumeric characters (must be within quotation marks); defined in ANSI/IEEE 488.2 as "String Response Data."	"Testing 1, 2, 3"

Special Characters

The Line Feed (LF) character (ASCII 10), and all characters in the range of ASCII 127-255 are defined as special characters. These characters are used in arbitrary block arguments only; using these characters in other parts of any command yields unpredictable results.

Abbreviating Commands, Queries, and Parameters

You can abbreviate most SCPI commands, queries, and parameters to an accepted short form. This manual shows these short forms as a combination of upper and lower case letters. The upper case letters indicate the accepted short form of a command. As shown in the following figure, you can create a short form by using only the upper case letters. The accepted short form and the long form are equivalent and request the same action of the instrument.

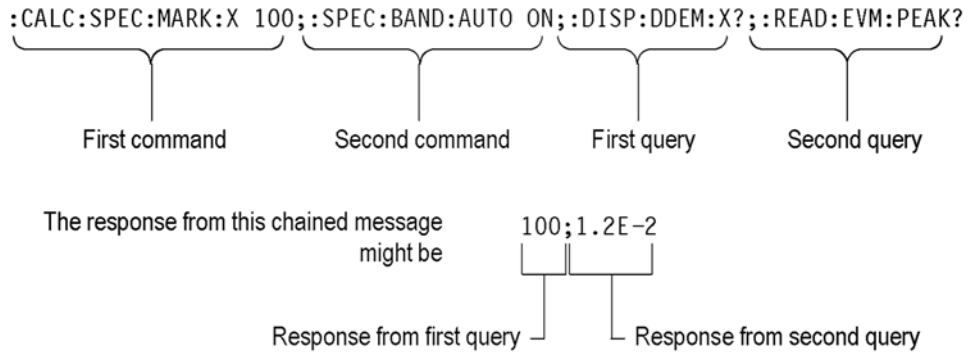


Example of abbreviating a command

NOTE: The numeric suffix of a command or query may be included in either the long form or short form; the analyzer will default to "1" if no suffix is used. In the above figure, the "3" of "MARKer3" indicates that the command is directed to Marker 3.

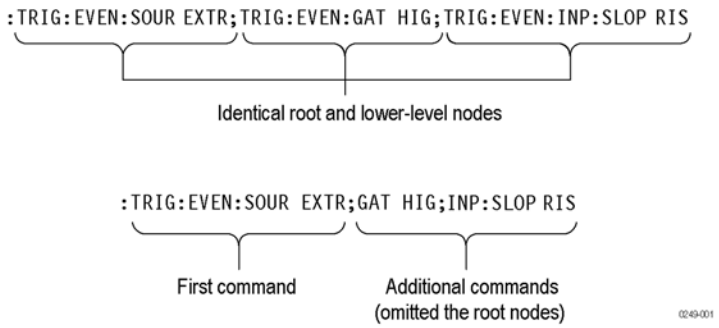
Chaining Commands and Queries

You can chain several commands or queries together into a single message. To create a chained message, first create a command or query, add a semicolon(;), and then add more commands or queries and semicolons until the message is complete. If the command following a semicolon is a root node, precede it with a colon (:). The following figure illustrates a chained message consisting of several commands and queries. The single chained message should end in a command or query, not a semicolon. Responses to any queries in your message are separated by semicolons.



Example of chaining commands and queries

If a command or query has the same root and lower-level nodes as the previous command or query, you can omit these nodes. In the following figure, the second command has the same root node (TRIGgerEVENt) as the first command, so these nodes can be omitted.



Example of omitting root and lower-level nodes in a chained message

Unit and SI Prefix

If the decimal numeric argument refers to amplitude, frequency, or time, you can express it using SI units instead of using the scaled explicit point input value format <NR3>. (SI units are units that conform to the Systeme International d'Unites standard.) For example, you can use the input format 200 mV or 1.0 MHz instead of 200.0E-3 or 1.0E+6, respectively, to specify voltage or frequency.

The following table lists the available units.

Available units

Symbol	Meaning
dB	decibel (relative amplitude)
dBm	decibel (absolute amplitude)
DEG	degree (phase)
Hz	hertz (frequency)
PCT	percent (%)
s	second (time)
V	volt

The available SI prefixes are shown in the following table.

Available SI prefixes

SI prefix	Z	A	F	P	N	U	M	K	MA*	G	T	PE	EX
Corresponding power	10^{-21}	10^{-18}	10^{-15}	10^{-12}	10^{-9}	10^{-6}	10^{-3}	10^{+3}	10^{+6}	10^{+9}	10^{+12}	10^{+15}	10^{+18}

* *When the unit is "Hz", "M" may be used instead of "MA" so that the frequency can be represented by "MHz"*

You can omit a unit in a command, but you must include the unit when using a SI prefix. For example, frequency of 15 MHz can be described as follows

15.0E6, 1.5E7Hz, 15000000, 15000000Hz, 15MHz, etc

("15M" is not allowed.)

Note that you can use either lower or upper case units and prefixes. The following examples have the same result, respectively.

170mHz, 170mHz, 170MHz, etc.

250mv, 250mV, 250MV, etc.

General Rules

Here are three general rules for using SCPI commands, queries, and parameters:

- You can use single (`' '`) or double (`" "`) quotation marks for quoted strings, but you cannot use both types of quotation marks for the same string.

  correct `"This string uses quotation marks correctly."`

  correct `'This string also uses quotation marks correctly.'`

  incorrect `"This string does not use quotation marks correctly.'`
- You can use upper case, lower case, or a mixture of both cases for all commands, queries, and parameters.

  `SENSE:SPECTRUM:FFT:LENGTH 1024`

is the same as

```
sense:spectrum:fft:length 1024
```

and

```
SENSE:spectrum:FFT:length 1024
```

NOTE: Literal strings (quoted) are case sensitive, for example, file names.

- No embedded spaces are allowed between or within nodes.

  correct `SENSE:SPECTRUM:FFT:LENGTH 1024`

  incorrect `SENSE: SPECTRUM: FFT: LEN GTH 1024`

IEEE 488.2 Common Commands

ANSI/IEEE Standard 488.2 defines the codes, formats, protocols, and usage of common commands and queries used on the interface between the controller and the instruments.The analyzer complies with this standard.

Command and Query Structure

The syntax for an IEEE 488.2 common command is an asterisk (*) followed by a command and, optionally, a space and parameter value. The syntax for an IEEE 488.2 common query is an asterisk (*) followed by a query and a question mark. All of the common commands and queries are listed in the last part of the **Syntax and Commands** section. The following are examples of common commands:

- *ESE 16
- *CLS

The following are examples of common queries

- *ESR
- *IDN

Constructed Mnemonics

Some header mnemonics specify one of a range of mnemonics. For example, a trace mnemonic can be either TRACe1, TRACe2, TRACe3, or TRACe4. You use these mnemonics in the command just as you do any other mnemonic. For example, there is a TRACe1:SPECtrum:FUNCtion command, and there is also a TRACe2:SPECtrum:FUNCtion command. In the command descriptions, this list of choices is abbreviated as TRACe<x>. The value of <x> is the upper range of valid suffixes. If the numeric suffix is omitted, the analyzer uses the default value of "1".

Constructed mnemonics

Symbol	Meaning
MARKer<x>	A marker specifier where <x> = 0, 1, 2, 3, or 4. Refer to Marker Mnemonics (on page 159)
RANGe<x>	A range specifier where <x> = 1 to 20. Refer to [SENSe]:SPURious Subgroup for details.
SPUR<x>	A spurious specifier where <x> = 1 to the number of spurious signals. Refer to FETCh|:READ:SPURious Subgroup for details, respectively.

Symbol	Meaning
TRACe<x>	A trace specifier where <x> = 0, 1, 2, 3, 4, or 5. Refer to TRACe Commands (on page 323) for details.

Command groups

This section lists the analyzer commands in two ways. It first presents them by functional groups. It then lists them alphabetically.The functional group list starts below.The alphabetical list provides more detail on each command. For the complete list of commands, see Command Descriptions (on page 342).

The analyzers conform to the Standard Commands for Programmable Instruments (SCPI) 1999.0 and IEEE Std 488.2-1987, except where noted.

Items followed by question marks are queries; items without question marks are commands. Some items in this section have a question mark in parentheses () in the command header section; this indicates that the item can be both a command and a query.

For the conventions of notation in this manual, refer to Command Syntax (on page 118) and following pages.

Functional groups

All commands are divided into groups as shown in the following table.

List of command group

Command group	Function
IEEE common	Conforms to the IEEE Std 488.2.
ABORt	Resets the trigger system and stops measurements.
CALCulate	Controls the markers and the search operations.
DISPlay	Controls the display of measurement results and waveforms.
FETCh	Retrieves the measurements from the last acquired data.
INITiate	Controls data acquisition.
INPut	Controls the characteristics of the signal input.
MMEMory	Provides mass storage capabilities for the analyzer.
OUTPut	Controls the characteristics of the signal output.
READ	Obtains the measurement results with acquiring data.
SENSe	Sets up detailed conditions for each measurement.
STATus	Controls the status and event registers.
SYSTem	Sets or queries system parameters for operation.

Command group	Function
TRACe	Controls trace activation and math operations.
UNIT	Specifies fundamental units for measurement.

Programming hints

Here are some basic tips for using commands:

- *Selecting a measurement item*

  Use Display commands to select or display the measurement view. [Example] `DISPlay:GENeral:MEASview:NEW SPECtrum` Displays the spectrum view on the screen.
- *Setting measurement parameters*

  Use Sense commands to set conditions for the measurement session. [Example] `SENSe:SPECtrum:FREQuency:CENTer 1.5GHz` Sets the center frequency to 1.5 GHz in the spectrum view.
- *Processing waveforms arithmetically*

  Use Trace commands for math operation on waveforms. [Example] `TRACe1:SPECtrum:FUNCtion AVERage` Averages the spectrum waveform.
- *Measuring with the markers*

  Use Calculate commands to measure some quantity using the markers. [Example] `CALCulate:SPECtrum:MARKer1:MAXimum` Positions the marker at the highest peak signal on the spectrum.
- *Scaling the waveform*

  Use Display commands to change the waveform portion on screen. [Example] `DISPlay:IQVTime:Y:SCALe 1.5` Sets the vertical range to 1.5 V in the IQ versus Time graph.

The following sections list the commands by group.

IEEE common commands

The IEEE 488.2 common commands have a "*" prefix.

Status and error commands

Command	Description
*CLS (on page 664)	Clears status.
*ESE (on page 1059)	Sets or queries the bits in the ESER register.
*ESR? (on page 1059)	Returns the contents of the SESR register.
*IDN? (on page 1479)	Returns the instrument identification code.
*OPC (on page 1544)	Synchronizes commands.
*OPT? (on page 1544)	Returns a list of options installed in your analyzer.
*RST	Returns the instrument settings to the factory defaults.
*SRE (on page 2482)	Sets or queries the bits in the SRER register.

Command	Description
*STB? (on page 2531)	Returns the contents of the SBR using the MSS bit.
*WAI (on page 2739)	Prevents the analyzer from executing further commands.

Obsolete commands

The following table lists obsolete commands and their replacements.

Obsolete and replacement commands

Obsolete commands	Replacement commands
[SENSe:]PULSe:ANALyze:MEASurement:TIME:AUTO	SENSe:PULSe:MEASview<y>:ANALyze:FDOMain:MEASurement:TIME:ALENgth
[SENSe:]PULSe:ANALyze:MEASurement:TIME:STARt	SENSe:PULSe:MEASview<y>:ANALyze:FDOMain:MEASurement:TIME:METHod

Obsolete commands	Replacement commands
[SENSe:]PULSe:ANALyze:MEASurement:TIME:STOP	SENSe:PULSe:MEASview<y>:ANALyze:FDOMain:MEASurement:TIME:RLENgth
	SENSe:PULSe:MEASview<y>:ANALyze:FDOMain:MEASurement:TIME:RLEVel
	SENSe:PULSe:MEASview<y>:ANALyze:FDOMain:MEASurement:TIME:STARt
	SENSe:PULSe:MEASview<y>:ANALyze:IRESponse:MEASurement:TIME:ALENgth
	SENSe:PULSe:MEASview<y>:ANALyze:IRESponse:MEASurement:TIME:METHod
	SENSe:PULSe:MEASview<y>:ANALyze:IRESponse:MEASurement:TIME:RLENgth
	SENSe:PULSe:MEASview<y>:ANALyze:IRESponse:MEASurement:TIME:RLEVel
	SENSe:PULSe:MEASview<y>:ANALyze:IRESponse:MEASurement:TIME:STARt
	SENSe:PULSe:MEASview<y>:ANALyze:OVERshoot:MEASurement:TIME:ALENgth
	SENSe:PULSe:MEASview<y>:ANALyze:OVERshoot:MEASurement:TIME:METHod
	SENSe:PULSe:MEASview<y>:ANALyze:OVERshoot:MEASurement:TIME:RLENgth
	SENSe:PULSe:MEASview<y>:ANALyze:OVERshoot:MEASurement:TIME:RLEVel
	SENSe:PULSe:MEASview<y>:ANALyze:OVERshoot:MEASurement:TIME:STARt
[SENSe]:PULSe:FREFerence:CHIRpbw(?)	SENSe:PULSe:MEASview<y>:FREFerence:CSLope

NOTE: <y> is the Measurement Index. Range: 1–8 (based on number of sources of connected instrument), where MEASview<y> keyword is optional and by default y is 1

Abort commands

Use the Abort commands to reset the trigger system and to stop measurements.

Abort commands

Command	Description
ABORt (on page 342)	Resets the trigger system and stops measurements.

Calculate commands

Use the Calculate commands to control the markers and the search operations.See the Index for commands not listed in the following table.

Calculate commands

Command	Description
CALCulate:MARKer:ADD (on page 458)	Adds a marker.
CALCulate:SYNC:MARKer<x>:MEASurement (on page 595)	Sets or queries the selected measurement to which the specified marker will be snapped to the peak of the measurement trace.
CALCulate:TXGain subcommand group	**Transmission Gain measurements**
CALculate:TXGain:MARKer<x>:DELTa:X? (on page 606)	Returns the delta marker frequency for the specified marker on the trace.
CALculate:TXGain:MARKer<x>:DELTa:Y? (on page 606)	Returns the delta marker amplitude for the specified marker on the trace.
CALculate:TXGain:MARKer<x>:MAXimum (on page 606)	Moves the specified marker to the highest peak on the trace.
CALculate:TXGain:MARKer<x>:PEAK:HIGHer (on page 607)	Moves the specified marker to the next peak higher in amplitude on the trace.
CALculate:TXGain:MARKer<x>:PEAK:LEFT (on page 607)	Moves the specified marker to the next peak to the left on the trace.
CALculate:TXGain:MARKer<x>:PEAK:LOWer (on page 608)	Moves the specified marker to the next peak lower in amplitude on the trace.
CALculate:TXGain:MARKer<x>:PEAK:RIGHt (on page 608)	Moves the specified marker to the next peak to the right on the trace.
CALculate:TXGain:MARKer<x>:TRACe (on page 609)	Sets or queries the trace on which the specified marker is placed in the Transmission Gain display.
CALculate:TXGain:MARKer<x>:X (on page 609)	Sets or queries the horizontal position for the specified marker on the trace.
CALculate:TXGain:MARKer<x>:Y? (on page 610)	Queries the marker amplitude of the specified marker on the trace.
CALCulate:BLUEtooth subcommand group	**Bluetooth measurements**
CALCulate:BLUEtooth:CONSte:MARKer<x>:DELTa:X[:TIME]? (on page 364)	Queries the delta marker time for the selected marker.
CALCulate:BLUEtooth:CONSte:MARKer<x>:FDEViation? (on page 364)	Queries the frequency deviation for the selected marker.

Command	Description
CALCulate:BLUETooth:CONSte:MARKer<x>:MAGNitude? (on page 365)	Queries the magnitude readout of the selected marker.
CALCulate:BLUEtooth:CONSte:MARKer<x>:MAXimum (on page 365)	Positions the selected marker at the symbol in the center of the analyzed time record.
CALCulate:BLUEtooth:CONSte:MARKer<x>:PEAK:LEFT (on page 366)	Moves the selected marker in the time domain to the next symbol number to the left, relative to the previous marker position.
CALCulate:BLUEtooth:CONSte:MARKer<x>:PEAK:RIGHt (on page 366)	Moves the selected marker in the time domain to the next symbol number to the right, relative to the previous marker position.
CALCulate:BLUETooth:CONSte:MARKer<x>:PHASe? (on page 367)	Queries the phase readout of the selected marker
CALCulate:BLUEtooth:CONSte:MARKer<x>:SYMBol? (on page 367)	Queries the symbol readout value for the selected marker.
CALCulate:BLUEtooth:CONSte:MARKer<x>:Trace (on page 368)	Sets or queries the Trace for the selected marker.
CALCulate:BLUETooth:CONSte:MARKer<x>:VALue? (on page 368)	Queries the value readout of the selected marker.
CALCulate:BLUEtooth:CONSte:MARKer<x>:X (on page 369)	Sets or queries the time position of the selected marker.
CALCulate:BLUEtooth:EDIagram:MARKer<x>:DELTa:X[:TIME]? (on page 369)	Returns the delta marker time for the selected marker.
CALCulate:BLUEtooth:EDIagram:MARKer<x>:DELTa:Y? (on page 370)	Returns the delta marker time for the selected marker.
CALCulate:BLUEtooth:EDIagram:MARKer<x>:MAXimum (on page 370)	Returns the delta marker amplitude for the selected markers.
CALCulate:BLUEtooth:EDIagram:MARKer<x>:PEAK:HIGHer (on page 371)	Moves the selected marker to the highest peak.
CALCulate:BLUEtooth:EDIagram:MARKer<x>:PEAK:LEFT (on page 371)	Moves the selected marker to the next peak to the left.
CALCulate:BLUEtooth:EDIagram:MARKer<x>:PEAK:LOWer (on page 372)	Moves the selected marker to the next peak lower in amplitude.
CALCulate:BLUEtooth:EDIagram:MARKer<x>:PEAK:RIGHt (on page 372)	Moves the selected marker to the next peak to the right.
CALCulate:BLUEtooth:EDIagram:MARKer<x>:TRACe (on page 373)	The command places the selected marker on the I or Q trace in the Bluetooth eye diagram display. The query returns the name of the trace where the current marker resides.

Command	Description
CALCulate:BLUEtooth:EDIagram:MARKer<x>:X[:TIME] (on page 373)	Sets or queries the horizontal position (time) of the selected marker.
CALCulate:BLUEtooth:EDIagram:MARKer<x>:Y? (on page 374)	Returns the vertical position of the selected marker.
CALCulate:BLUEtooth:FDVTIME:MARKer<x>:DELTa:X[:TIME]? (on page 374)	Queries the delta marker time for the selected marker.
CALCulate:BLUEtooth:FDVTIME:MARKer<x>:DELTa:Y? (on page 375)	Returns the delta marker amplitude for the selected marker .
CALCulate:BLUEtooth:FDVTime:MARKer<x>:MAXimum (on page 376)	Moves the selected marker to the highest peak on the trace.
CALCulate:BLUEtooth:FDVTime:MARKer<x>:PEAK:HIGHer (on page 376)	Moves the selected marker to the next peak higher in amplitude.
CALCulate:BLUEtooth:FDVTime:MARKer<x>:PEAK:LEFT (on page 377)	Moves the selected marker to the next peak to the left on the trace.
CALCulate:BLUEtooth:FDVTime:MARKer<x>:PEAK:LOWer (on page 377)	Moves the selected marker to the next peak lower in amplitude.
CALCulate:BLUEtooth:FDVTime:MARKer<x>:PEAK:RIGHT (on page 378)	Moves the selected marker to the next peak to the right on the trace.
CALCulate:BLUEtooth:FDVTIME:MARKer<x>:X[:TIME] (on page 378)	Sets or queries the marker time.
CALCulate:BLUEtooth:FDVTIME:MARKer<x>:Y (on page 379)	Sets or queries the marker amplitude.
CALCulate:BOBW:MARKer<x>:DELTa:X? (on page 379)	Returns the delta marker frequency for the selected marker.
CALCulate:BOBW:MARKer<x>:DELTa:Y? (on page 380)	Returns the delta marker frequency for the selected marker on the Bluetooth 20 dB bandwidth measurement.
CALCulate:BOBW:MARKer<x>:MAXimum (on page 380)	Positions the selected marker at the maximum value of the display.
CALCulate:BOBW:MARKer<x>:PEAK:HIGHer (on page 381)	Moves the selected marker to the next peak higher in amplitude.
CALCulate:BOBW:MARKer<x>:PEAK:LEFT (on page 381)	Moves the selected marker in the time domain to the left, relative to the previous marker position on the trace.
CALCulate:BOBW:MARKer<x>:PEAK:LOWer (on page 381)	Moves the selected marker to the next peak lower in amplitude.
CALCulate:BOBW:MARKer<x>:PEAK:RIGHt (on page 382)	Moves the selected marker in the time domain to the right, relative to the previous marker position on the trace.

Command	Description
CALCulate:BOBW:MARKer<x>[:SET]:CENTer (on page 382)	Sets the center frequency to the value at the marker position.
CALCulate:BOBW:MARKer<x>:X (on page 383)	Sets or queries the time position of the selected marker.
CALCulate:BOBW:MARKer<x>:Y? (on page 383)	Returns the amplitude position of the selected marker.
CALCulate:DPX:MARKer subcommand group	**DPX power density readings**
CALCulate:DPX:MARKer<x>:POWer:DENSity? (on page 400)	Returns the power density of the specified marker.
CALCulate:DPX:MARKer<x>:POWer:IDensity? (on page 401)	Returns the integrated power density reading between the MR (reference marker) and the selected marker.
CALCulate:DPX:MARKer<x>:POWer:INTEgrated? (on page 401)	Returns the integrated power reading between the MR (reference marker) and the selected marker.
CALCulate:MARKer:MODE (on page 462)	Sets or queries the marker mode.
CALCulate:NOISe subcommand group	**Noise and Gain measurements**
CALCulate:NOISe:FIGure:MARKer<x>:DELTa:X? (on page 472)	Queries the frequency difference between MR (reference marker) and the selected marker
CALCulate:NOISe:FIGure:MARKer<x>:DELTa:Y? (on page 473)	Queries the power difference value between MR (reference marker) and the selected marker.
CALCulate:NOISe:FIGure:MARKer<x>:MAXimum (on page 473)	Places the selected marker at the maximum value on the peak of the trace.
CALCulate:NOISe:FIGure:MARKer<x>:PEAK:HIGHer (on page 474)	Places the selected marker at the next highest peak on the trace.
CALCulate:NOISe:FIGure:MARKer<x>:PEAK:LEFT (on page 474)	Places the selected marker at the next peak on the trace to the left of the selected marker
CALCulate:NOISe:FIGure:MARKer<x>:PEAK:LOWer (on page 475)	Places the selected marker at the next lowest peak on the trace.
CALCulate:NOISe:FIGure:MARKer<x>:PEAK:RIGHt (on page 475)	Places the selected marker at the next peak on the trace to the right of the selected marker.
CALCulate:NOISe:FIGure:MARKer<x>:TRACe (on page 476)	Sets or queries which trace the specified marker is attached to.
CALCulate:NOISe:FIGure:MARKer<x>:X? (on page 477)	Queries the frequency of the specified marker.
CALCulate:NOISe:FIGure:MARKer<x>:Y? (on page 477)	Queries the vertical position (dB) of the specified marker.

Command	Description
CALCulate:NOISe:GAIN:MARKer<x>:DELTa:X? (on page 478)	Query returns the frequency difference between MR (reference marker) and the selected marker.
CALCulate:NOISe:GAIN:MARKer<x>:DELTa:Y? (on page 478)	Query returns the power difference value between MR (reference marker) and the selected marker.
CALCulate:NOISe:GAIN:MARKer<x>:MAXimum? (on page 479)	Places the selected marker at the maximum value on the peak of the trace.
CALCulate:NOISe:GAIN:MARKer<x>:PEAK:HIGHer? (on page 479)	Places the selected marker at the next highest peak on the trace.
CALCulate:NOISe:GAIN:MARKer<x>:PEAK:LEFT? (on page 480)	Places the selected marker at the next peak on the trace to the left of the selected marker.
CALCulate:NOISe:GAIN:MARKer<x>:PEAK:LOWer? (on page 480)	Places the selected marker at the next lowest peak on the trace.
CALCulate:NOISe:GAIN:MARKer<x>:PEAK:RIGHt? (on page 481)	Places the specified marker on a specified trace. The query form returns the trace on which the marker resides.
CALCulate:NOISe:GAIN:MARKer<x>:TRACe (on page 481)	Sets or queries which trace the specified marker is attached to.
CALCulate:NOISe:GAIN:MARKer<x>:X? (on page 482)	Queries the horizontal position of the specified marker.
CALCulate:NOISe:GAIN:MARKer<x>:Y? (on page 483)	Queries the vertical position of the specified marker.
CALCulate:NOISe:TEMPerature:MARKer<x>:DELTa:X? (on page 483)	Returns the frequency difference between MR (reference marker) and the selected marker.
CALCulate:NOISe:TEMPerature:MARKer<x>:DELTa:Y? (on page 484)	Returns the temperature difference between MR (reference marker) and the selected marker.
CALCulate:NOISe:TEMPerature:MARKer<x>:MAXimum? (on page 484)	Places the selected marker at the maximum value on the peak of the trace.
CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:HIGHer? (on page 485)	Places the selected marker at the highest peak on the trace.
CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:LEFT? (on page 485)	Places the selected marker at the next peak on the trace to the left of the selected marker.
CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:LOWer? (on page 486)	Places the selected marker at the next lowest peak on the trace.
CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:RIGHt? (on page 486)	This command places the specified marker on a specified trace. The query form returns the trace on which the marker resides.
CALCulate:NOISe:TEMPerature:MARKer<x>:TRACe (on page 487)	Sets or queries which trace the specified marker is attached to.

Command	Description
CALCulate:NOISe:TEMPerature:MARKer<x>:X? (on page 488)	Queries the frequency of the specified marker.
CALCulate:NOISe:TEMPerature:MARKer<x>:Y? (on page 488)	Queries the temperature (Kelvin) at the specified marker.
CALCulate:NOISe:YFACtor:MARKer<x>:DELTa:X? (on page 489)	Queries the frequency difference between MR (reference marker) and the selected marker.
CALCulate:NOISe:YFACtor:MARKer<x>:DELTa:Y? (on page 489)	Queries the power difference value between MR (reference marker) and the selected marker.
CALCulate:NOISe:YFACtor:MARKer<x>:MAXimum (on page 490)	Places the selected marker at the maximum value on the peak of the trace.
CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:LEFT? (on page 491)	Places the selected marker at the next peak to the left on the trace.
CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:LOWer? (on page 491)	Places the selected marker at the next lowest peak on the trace.
CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:RIGHt? (on page 492)	Places the specified marker on a specified trace. The query form returns the trace on which the marker resides.
CALCulate:NOISe:YFACtor:MARKer<x>:TRACe (on page 492)	Sets or queries which trace the specified marker is attached to.
CALCulate:NOISe:YFACtor:MARKer<x>:X? (on page 493)	Queries the frequency of the specified marker.
CALCulate:NOISe:YFACtor:MARKer<x>:Y? (on page 493)	This command queries the vertical position (dB) of the specified marker.
CALCulate:P25 subcommand group	**P25 analysis**
CALCulate:P25:CONSte:MARKer<x>:DELTa:X[:TIME]? (on page 509)	Returns the delta marker time for the selected marker.
CALCulate:P25:CONSte:MARKer<x>:FDEViation? (on page 510)	Queries the frequency deviation of the selected marker.
CALCulate:P25:CONSte:MARKer<x>:MAGNitude? (on page 510)	Queries the frequency deviation (for Freq Dev trace types) or magnitude (for IQ trace types) of the selected marker.
CALCulate:P25:CONSte:MARKer<x>:MAXimum (on page 511)	Moves the selected marker to the highest peak on the trace.
CALCulate:P25:CONSte:MARKer<x>:PEAK:LEFT (on page 511)	Moves the selected marker to the next peak to the left on the trace.
CALCulate:P25:CONSte:MARKer<x>:PEAK:RIGHt (on page 511)	Moves the selected marker to the next peak to the right on the trace.

Command	Description
CALCulate:P25:CONSte:MARKer<x>:PHASe? (on page 512)	Queries the phase of the selected marker.
CALCulate:P25:CONSte:MARKer<x>:SYMBol? (on page 512)	Queries the readout symbol of the selected marker.
CALCulate:P25:CONSte:MARKer<x>:Trace (on page 513)	Places the selected marker on the Demodulated I and Q versus Time trace. The query returns the name of the trace on which the marker resides.
CALCulate:P25:CONSte:MARKer<x>:VALue? (on page 513)	Queries the readout value of the specified marker.
CALCulate:P25:CONSte:MARKer<x>:X (on page 514)	Sets or queries the horizontal position of the selected marker.
CALCulate:P25:EDIagram:MARKer<x>:DELTa:X [:TIME]? (Query Only)	Returns the delta marker time for the selected marker.
CALCulate:P25:EDIagram:MARKer<x>:DELTa:Y? (on page 515)	Returns the delta marker time for the selected marker.
CALCulate:P25:EDIagram:MARKer<x>:MAXimum (on page 515)	Moves the selected marker to the highest peak.
CALCulate:P25:EDIagram:MARKer<x>:PEAK:HIGHer (on page 516)	Moves the selected marker to the next highest peak.
CALCulate:P25:EDIagram:MARKer<x>:PEAK:LEFT (on page 517)	Moves the selected marker to the next peak to the left .
CALCulate:P25:EDIagram:MARKer<x>:PEAK:LOWer (on page 517)	Moves the selected marker to the next lowest peak.
CALCulate:P25:EDIagram:MARKer<x>:PEAK:RIGHt (on page 518)	Moves the selected marker to the next peak to the right .
CALCulate:P25:EDIagram:MARKer<x>:TRACe (on page 519)	Sets the selected marker on the I or Q trace, or queries the trace that the specified marker is on.
CALCulate:P25:EDIagram:MARKer<x>:X (on page 519)	Sets or queries the horizontal position of the selected marker.
CALCulate:P25:EDIagram:MARKer<x>:Y (on page 520)	Sets or queries the vertical position of the selected marker.
CALCulate:P25:PVTime:MARKer<x>:DELTa:X[:TIME]? (on page 520)	Returns the delta marker time (seconds) for the selected marker.
CALCulate:P25:PVTime:MARKer<x>:DELTa:Y[:TIME]? (on page 521)	Returns the Delta marker amplitude (dB) for the selected marker.
CALCulate:P25:PVTime:MARKer<x>:Maximum (on page 521)	Moves the selected marker to the highest peak on the trace.

Command	Description
CALCulate:P25:PVTime:MARKer<x>:PEAK:HIGHer (on page 522)	Moves the selected marker to the next peak higher in amplitude on the trace.
CALCulate:P25:PVTime:MARKer<x>:PEAK:LEFT (on page 523)	Moves the selected marker to the next peak to the left on the trace.
CALCulate:P25:PVTime:MARKer<x>:PEAK:LOWer (on page 523)	Moves the selected marker to the next peak lower in amplitude on the trace.
CALCulate:P25:PVTime:MARKer<x>:PEAK:RIGHt (on page 524)	Moves the selected marker to the next peak to the right on the trace.
CALCulate:P25:PVTime:MARKer<x>:X (on page 524)	Sets or queries the horizontal position of the selected marker.
CALCulate:P25:PVTime:MARKer<x>:Y (on page 525)	Sets or queries the vertical position of the selected marker.
CALCulate: Marker subgroup	**Marker position and values**
CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:POWer:DENSity? (on page 586)	Returns the power density of the specified marker.
CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:POWer:IDENSity (on page 587)	Returns the integrated power density reading between the MR (reference marker) and the selected marker.
CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:POWer:INTEgrated? (on page 588)	Returns the integrated power reading between the MR (reference marker) and the selected marker.
CALCulate:MARKer:AOFF (on page 459)	Turns off all markers.
CALCulate:MARKer:DELete (on page 459)	Deletes the last marker added.
CALCulate:MARKer:DENSity:EXCursion (on page 460)	Sets or queries the minimum excursion of DPX signal density.
CALCulate:MARKer:DENSity:SMOothing (on page 460)	Sets or queries the number of pixels squared for smoothing the density.
CALCulate:MARKer:DENSity:THReshold (on page 461)	Sets or queries the threshold of DPX signal density to detect peaks.
CALCulate:MARKer:DRAG:SEARch:STATe (on page 461)	Enables or disables peak searching when dragging a marker in a plot.
CALCulate:MARKer:MODE (on page 461)	Sets or queries the marker mode.
CALCulate:MARKer:NOISe:MODE (on page 462)	Sets or queries the marker noise mode.
CALCulate:MARKer:PEAK:EXCursion (on page 463)	Sets or queries the minimum excursion level.

Command	Description
CALCulate:MARKer:PEAK:THReshold (on page 463)	Sets or queries the threshold level to detect peaks.
CALCulate:SEARch subgroup	**Calculate search commands**
CALCulate:SEARch:LIMit:FAIL? (on page 555)	Queries whether the waveform cuts across the limit or not.
CALCulate:SEARch:LIMit:MATCh:BEEP[:STATe] (on page 555)	Sets or queries whether to beep when a match occurs.
CALCulate:SEARch:LIMit:MATCh:SACQuire[:STATe] (on page 555)	Sets or queries whether to stop acquiring data on match.
CALCulate:SEARch:LIMit:MATCh:SDATa[:STATe] (on page 556)	Sets or queries whether to save the acquisition data automatically.
CALCulate:SEARch:LIMit:MATCh:SPICture[:STATe] (on page 556)	Sets or queries whether to save the whole screen automatically.
CALCulate:SEARch:LIMit:MATCh:STRace[:STATe] (on page 557)	Sets or queries whether to save the spectrum trace automatically.
CALCulate:SEARch:LIMit:OPERation (on page 558)	Sets or queries the limit operation in the search function.
CALCulate:SEARch:LIMit:OPERation:FEED (on page 559)	Sets or queries the data flow to be fed in the search operation.
CALCulate:SEARch:LIMit:OPERation:MASK:LOAD (on page 559)	Loads the limit mask from a specified file for the search operation.
CALCulate:SEARch:LIMit:OPERation:MASK:STORE (on page 560)	Stores the limit mask to a specified file for the search operation.
CALCulate:[:SEARch|:TEST]:LIMit[:STATe]:MATCh:FILE:LOCation (on page 343)	Sets or queries the limit value in the search operation.
CALCulate:SEARch:LIMit:REPort:DATA? (on page 561)	Returns the frequency range(s) that satisfy the search condition.
CALCulate:SEARch:LIMit:REPort:POINts? (on page 561)	Returns the number of range(s) that satisfy the search condition.
CALCulate:SEARch:LIMit:STATe (on page 562)	Sets or queries whether to enable or disable the search function.
CALCulate:ACPower subgroup	**Channel power and ACPR measurement**
CALCulate:ACPower:MARKer<x>:DELTa:X? (on page 344)	Returns the delta marker frequency for the selected marker.
CALCulate:ACPower:MARKer<x>:DELTa:Y? (on page 344)	Returns the delta marker amplitude for the selected marker.

Command	Description
CALCulate:ACPower:MARKer<x>:MAXimum (on page 345)	Moves the marker to the highest peak on the trace.
CALCulate:ACPower:MARKer<x>:PEAK:LEFT (on page 345)	Moves the marker to the next peak to the left on the trace.
CALCulate:ACPower:MARKer<x>:PEAK:RIGHt (on page 345)	Moves the marker to the next peak to the right on the trace.
CALCulate:ACPower:MARKer<x>:X (on page 346)	Sets or queries the horizontal position of the marker.
CALCulate:ACPower:MARKer<x>:Y? (on page 346)	Queries the marker amplitude of the selected marker.
CALCulate:{AM|FM|PM} subgroup (Option 21 only)	**AM/FM/PM measurement**
CALCulate:{AM|FM|PM}:MARKer<x>:DELTa:X? (on page 347)	Returns the delta marker time for the selected marker.
CALCulate:{AM|FM|PM}:MARKer<x>:DELTa:Y? (on page 347)	Returns the delta marker amplitude for the selected marker.
CALCulate:{AM|FM|PM}:MARKer<x>:MAXimum (on page 348)	Moves the marker to the highest peak on the trace.
CALCulate:{AM|FM|PM}:MARKer<x>:PEAK:HIGHer (on page 348)	Moves the marker to the next peak higher in amplitude.
CALCulate:{AM|FM|PM}:MARKer<x>:PEAK:LEFT (on page 349)	Moves the marker to the next peak to the left on the trace.
CALCulate:{AM|FM|PM}:MARKer<x>:PEAK:LOWer (on page 349)	Moves the marker to the next peak lower in amplitude.
CALCulate:{AM|FM|PM}:MARKer<x>:PEAK:RIGHt (on page 350)	Moves the marker to the next peak to the right on the trace.
CALCulate:{AM|FM|PM}:MARKer<x>:X (on page 350)	Sets or queries the horizontal position (time) of the marker.
CALCulate:{AM|FM|PM}:MARKer<x>:Y? (on page 351)	Queries the marker amplitude of the selected marker.
CALCulate:AUDio subgroup	**Audio measurements**
CALCulate:AUDio:HARMonic:HNUMber (on page 351)	Sets or queries the number of harmonics.
CALCulate:AUDio:HARMonic:NHNumber (on page 352)	Sets or queries the number of non-harmonics.
CALCulate:AUDio:NHARmonic:EXCursion (on page 352)	Sets or queries the non-harmonic excursion value.

Command	Description
CALCulate:AUDio:NHARmonic:IGNore (on page 352)	Sets or queries the ignore region of the non-harmonics.
CALCulate:AUDio:NHARmonic:THReshold (on page 353)	Sets or queries the non-harmonic threshold value.
CALCulate:AUDio:SPECtrum:MARKer<x>:DELTa:X? (on page 353)	Returns the delta marker frequency for the selected marker on the spectrum trace.
CALCulate:AUDio:SPECtrum:MARKer<x>:DELTa:Y? (on page 354)	Returns the delta marker amplitude for the selected marker on the spectrum trace.
CALCulate:AUDio:SPECtrum:MARKer<x>:MAXimum (on page 354)	Moves the selected marker to the highest peak on the spectrum trace.
CALCulate:AUDio:SPECtrum:MARKer<x>:PEAK:HIGHer (on page 355)	Moves the selected marker to the next peak higher in amplitude on the spectrum trace.
CALCulate:AUDio:SPECtrum:MARKer<x>:PEAK:LEFT (on page 355)	Moves the selected marker to the next peak to the left on the spectrum trace.
CALCulate:AUDio:SPECtrum:MARKer<x>:PEAK:LOWer (on page 356)	Moves the selected marker to the next peak lower in amplitude on the spectrum trace.
CALCulate:AUDio:SPECtrum:MARKer<x>:PEAK:RIGHt (on page 356)	Moves the selected marker to the next peak to the right on the spectrum trace.
CALCulate:AUDio:SPECtrum:MARKer<x>:TRACe (on page 356)	Sets or queries the trace on which the specified marker is placed in the spectrum measurement.
CALCulate:AUDio:SPECtrum:MARKer<x>:X (on page 357)	Sets or queries the horizontal position of the selected marker on the spectrum trace.
CALCulate:AUDio:SPECtrum:MARKer<x>:Y? (on page 357)	Queries the marker amplitude of the selected marker on the spectrum trace.
CALCulate:AVTime subgroup	**Frequency versus Time measurement**
CALCulate:AVTime[:MEASview<y>]:MARKer<x>:DELTa:X? (on page 358)	Returns the delta marker time for the selected marker.
CALCulate:AVTime[:MEASview<y>]:MARKer<x>:DELTa:Y? (on page 359)	Returns the delta marker amplitude for the selected marker.
CALCulate:AVTime[:MEASview<y>]:MARKer<x>:MAXimum (on page 359)	Moves the marker to the highest peak on the trace.
CALCulate:AVTime[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (on page 360)	Moves the marker to the next peak higher in amplitude.
CALCulate:AVTime[:MEASview<y>]:MARKer<x>:PEAK:LEFT (on page 360)	Moves the marker to the next peak to the left on the trace.

Command	Description
CALCulate:AVTime[:MEASview<y>]:MARKer<x>:PEAK:LOWer (on page 361)	Moves the marker to the next peak lower in amplitude.
CALCulate:AVTime[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (on page 361)	Moves the marker to the next peak to the right on the trace.
CALCulate:AVTime[:MEASview<y>]:MARKer<x>:TRACe (on page 362)	Sets or queries the trace on which the marker is placed.
CALCulate:AVTime[:MEASview<y>]:MARKer<x>:X (on page 362)	Sets or queries the horizontal position of the marker.
CALCulate:AVTime[:MEASview<y>]:MARKer<x>:Y? (on page 363)	Queries the marker amplitude of the selected marker.
CALCulate:CONSte subgroup (Option 21 only)	**Constellation measurement**
CALCulate:CONSte[:MEASview<y>]:MARKer<x>:DELTa:X[:TIME]? (on page 384)	Returns the delta marker time for the selected marker on the Constellation trace.
CALCulate:CONSte[:MEASview<y>]:MARKer<x>:FDEViation? (on page 384)	Queries the frequency deviation of the selected marker in the Constellation measurement for an FSK modulated signal.
CALCulate:CONSte[:MEASview<y>]:MARKer<x>:MAGNitude? (on page 385)	Queries the magnitude readout of the selected marker in the Constellation measurement.
CALCulate:CONSte[:MEASview<y>]:MARKer<x>:MAXimum (on page 386)	Positions the selected marker at the symbol in the center of the time record.
CALCulate:CONSte[:MEASview<y>]:MARKer<x>:PEAK:LEFT (on page 386)	Moves the selected marker in the time domain to the next lower symbol number, relative to the previous marker position.
CALCulate:CONSte[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (on page 387)	Moves the selected marker in the time domain to the next higher symbol number, relative to the previous marker position.
CALCulate:CONSte[:MEASview<y>]:MARKer<x>:PHASe? (on page 387)	Queries the phase readout of the selected marker in the Constellation measurement.
CALCulate:CONSte[:MEASview<y>]:MARKer<x>:SYMBol? (on page 388)	Queries the symbol readout of the selected marker in the Constellation measurement.
CALCulate:CONSte[:MEASview<y>]:MARKer<x>:VALue? (on page 389)	Queries the value readout of the selected marker in the Constellation measurement.
CALCulate:CONSte[:MEASview<y>]:MARKer<x>:X (on page 389)	Sets or queries the time position of the selected marker on the Constellation trace.
CALCulate:DIQVtime subgroup (Option 21 only)	**Demodulated I&Q versus Time measurement**
CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:DELTa:X[:TIME]? (on page 390)	Returns the delta marker time for the selected marker.
CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:DELTa:Y? (on page 391)	Returns the delta marker amplitude for the selected marker.

Command	Description
CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:MAXimum (on page 391)	Moves the selected marker to the highest peak on the trace.
CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (on page 392)	Moves the selected marker to the next peak higher in amplitude.
CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:PEAK:LEFT (on page 393)	Moves the selected marker to the next peak to the left on the trace.
CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:PEAK:LOWer (on page 393)	Moves the selected marker to the next peak lower in amplitude.
CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (on page 394)	Moves the selected marker to the next peak to the right on the trace.
CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:TRACe (on page 394)	Places the selected marker on the Demodulated I&Q versus Time trace.
CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:X[:TIME] (on page 395)	Sets or queries the horizontal position (time) of the selected marker.
CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:Y? (on page 396)	Queries the marker amplitude of the selected marker.
CALCulate:DPX subgroup	**DPX spectrum measurement**
CALCulate:DPX:MARKer<x>:DELTa:Y? (on page 397)	Returns the delta marker amplitude for the selected marker.
CALCulate:DPX:MARKer<x>:MAXimum (on page 398)	Moves the marker to the highest peak on the trace.
CALCulate:DPX:MARKer<x>:PEAK:HIGHer (on page 398)	Moves the marker to the next peak higher in amplitude.
CALCulate:DPX:MARKer<x>:PEAK:LEFT (on page 399)	Moves the marker to the next peak to the left on the trace.
CALCulate:DPX:MARKer<x>:PEAK:LOWer (on page 399)	Moves the marker to the next peak lower in amplitude.
CALCulate:DPX:MARKer<x>:PEAK:RIGHt (on page 400)	Moves the marker to the next peak to the right on the trace.
CALCulate:DPX:MARKer<x>[:SET]:CENTer (on page 402)	Sets the center frequency to the marker frequency.
CALCulate:DPX:MARKer<x>:TRACe (on page 402)	Sets or queries the trace to attach the marker to.
CALCulate:DPX:MARKer<x>:X:AMPLitude (on page 403)	Sets or queries the amplitude position of the marker.

Command	Description
CALCulate:DPX:MARKer<x>:X[:FREQuency] (on page 403)	Sets or queries the frequency position of the marker.
CALCulate:DPX:MARKer<x>:X:PHASe (on page 404)	Sets or queries the phase position of the selected marker.
CALCulate:DPX:MARKer<x>:X:TIME (on page 404)	Sets or queries the time position of the selected marker.
CALCulate:DPX:MARKer<x>:Y? (on page 405)	Queries the marker amplitude of the selected marker.
CALCulate:EDIagram subgroup (Option 21 only)	**Eye diagram**
CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:DELTa:X[:TIME]?	Returns the delta marker time for the selected marker on the eye diagram trace.
CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:DELTa:Y? (on page 406)	Returns the delta marker amplitude for the selected marker on the eye diagram trace.
CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:MAXimum	Moves the selected marker to the highest peak on the eye diagram trace.
CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (on page 407)	Moves the selected marker to the next peak higher in amplitude on the eye diagram trace.
CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:PEAK:LEFT (on page 408)	Moves the selected marker to the next peak to the left on the eye diagram trace.
CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:PEAK:LOWer (on page 408)	Moves the selected marker to the next peak lower in amplitude on the eye diagram trace.
CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (on page 409)	Moves the selected marker to the next peak to the right on the eye diagram trace.
CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:TRACe (on page 410)	Places the selected marker on the I or Q trace in the eye diagram display.
CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:X[:TIME] (on page 410)	Sets or queries the horizontal position (time) of the selected marker in the eye diagram measurement.
CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:Y? (on page 411)	Queries the vertical position of the selected marker in the eye diagram measurement.
CALCulate:EVM subgroup (Option 21 only)	**EVM versus Time measurement**
CALCulate:EVM[:MEASview<y>]:MARKer<x>:DELTa:X? (on page 416)	Returns the delta marker amplitude for the selected marker in the EVM versus Time measurement.
CALCulate:EVM[:MEASview<y>]:MARKer<x>:DELTa:Y? (on page 417)	Returns the delta marker amplitude for the selected marker in the EVM versus Time measurement.
CALCulate:EVM[:MEASview<y>]:MARKer<x>:MAXimum (on page 418)	Moves the selected marker to the highest peak on the trace in the EVM versus Time measurement.

Command	Description
CALCulate:EVM[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (on page 418)	Moves the selected marker to the next peak higher in amplitude on the EVM versus Time trace.
CALCulate:EVM[:MEASview<y>]:MARKer<x>:PEAK:LEFT (on page 419)	Moves the selected marker to the next peak to the left on the trace in the EVM versus Time measurement.
CALCulate:EVM[:MEASview<y>]:MARKer<x>:PEAK:LOWer (on page 419)	Moves the selected marker to the next peak lower in amplitude on the EVM versus Time trace.
CALCulate:EVM[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (on page 420)	Moves the selected marker to the next peak to the right on the trace in the EVM versus Time measurement.
CALCulate:EVM[:MEASview<y>]:MARKer<x>:X (on page 421)	Sets or queries the horizontal position of the selected marker in the EVM versus Time measurement.
CALCulate:EVM[:MEASview<y>]:MARKer<x>:Y? (on page 421)	Queries the marker amplitude of the selected marker in the EVM versus Time measurement.
CALCulate:FDVTime subgroup (Option 21 only)	**Frequency deviation versus Time measurement**
CALCulate:FDVTime[:MEASview<y>]:MARKer<x>:DELTa:X[:TIME]? (on page 422)	Returns the delta marker time for the selected marker on the Frequency deviation versus Time trace.
CALCulate:FDVTime[:MEASview<y>]:MARKer<x>:DELTa:Y? (on page 422)	Returns the delta marker amplitude for the selected marker in the Frequency deviation versus Time measurement.
CALCulate:FDVTime[:MEASview<y>]:MARKer<x>:MAXimum (on page 423)	Moves the selected marker to the highest peak on the trace in the Frequency deviation versus Time measurement.
CALCulate:FDVTime[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (on page 424)	Moves the marker to the next peak higher in amplitude.
CALCulate:FDVTime[:MEASview<y>]:MARKer<x>:PEAK:LEFT (on page 424)	Moves the marker to the next peak to the left on the trace.
CALCulate:FDVTime[:MEASview<y>]:MARKer<x>:PEAK:LOWer (on page 425)	Moves the selected marker to the next peak lower in amplitude on the Frequency deviation versus Time trace.
CALCulate:FDVTime[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (on page 425)	Moves the selected marker to the next peak to the right on the trace in the Frequency deviation versus Time measurement.
CALCulate:FDVTime[:MEASview<y>]:MARKer<x>:X[:TIME] (on page 426)	Sets or queries the horizontal position of the selected marker in the Frequency deviation versus Time measurement.
CALCulate:FDVTime[:MEASview<y>]:MARKer<x>:Y? (on page 427)	Queries the marker amplitude of the selected marker in the Frequency deviation versus Time measurement.
CALCulate:{FSETtling|PSETtling} subgroup	**Frequency and Phase Settling measurements**
CALCulate:{FSETtling|PSETtling}:MARKer<x>:DELTa:X? (on page 427)	Returns the delta marker time for the selected marker in the Frequency and Phase Settling displays.
CALCulate:{FSETtling|PSETtling}:MARKer<x>:DELTa:Y? (on page 428)	Returns the delta marker frequency for the selected marker in the Frequency and Phase Settling displays.

Command	Description
CALCulate:{FSETtling|PSETtling}:MARKer<x>:MAXimum (on page 428)	Moves the selected marker to the highest peak on the trace in the Frequency or Phase Settling display.
CALCulate:{FSETtling|PSETtling}:MARKer<x>:PEAK:HIGHer (on page 429)	Moves the selected marker to the next peak higher in amplitude on the Frequency or Phase Settling trace.
CALCulate:{FSETtling|PSETtling}:MARKer<x>:PEAK:LEFT (on page 429)	Moves the selected marker to the next peak to the left on the trace in the Frequency or Phase Settling display.
CALCulate:{FSETtling|PSETtling}:MARKer<x>:PEAK:LOWer (on page 430)	Moves the selected marker to the next peak lower in amplitude on the Frequency or Phase Settling trace.
CALCulate:{FSETtling|PSETtling}:MARKer<x>:PEAK:RIGHt (on page 430)	Moves the selected marker to the next peak to the right on the trace in the Frequency or Phase Settling display.
CALCulate:{FSETtling|PSETtling}:MARKer<x>:TRACe (on page 431)	Sets or queries the trace on which the specified marker is placed in the spectrum measurement
CALCulate:{FSETtling|PSETtling}:MARKer<x>:X (on page 431)	Sets or queries the horizontal position of the selected marker in the Frequency and Phase Settling display.
CALCulate:{FSETtling|PSETtling}:MARKer<x>:Y? (on page 432)	Queries the marker amplitude of the selected marker in the Frequency and Phase Settling display.
CALCulate:FVTime subgroup	**Frequency versus Time measurement**
CALCulate:FVTime[:MEASview<y>]:MARKer<x>:DELTa:X? (on page 432)	Returns the delta marker time for the selected marker.
CALCulate:FVTime[:MEASview<y>]:MARKer<x>:DELTa:Y? (on page 433)	Returns the delta marker frequency for the selected marker.
CALCulate:FVTime[:MEASview<y>]:MARKer<x>:MAXimum (on page 433)	Moves the marker to the highest peak on the trace.
CALCulate:FVTime[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (on page 434)	Moves the marker to the next peak higher in amplitude.
CALCulate:FVTime[:MEASview<y>]:MARKer<x>:PEAK:LEFT (on page 434)	Moves the marker to the next peak to the left on the trace.
CALCulate:FVTime[:MEASview<y>]:MARKer<x>:PEAK:LOWer (on page 435)	Moves the marker to the next peak lower in amplitude.
CALCulate:FVTime[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (on page 435)	Moves the marker to the next peak to the right on the trace.
CALCulate:FVTime[:MEASview<y>]:MARKer<x>:TRACe	Sets or queries the trace on which the specified marker is placed in the Frequency versus Time measurement.
CALCulate:FVTime[:MEASview<y>]:MARKer<x>:X (on page 436)	Sets or queries the horizontal position of the marker.

Command	Description
CALCulate:FVTime[:MEASview<y>]:MARKer<x>:Y? (on page 437)	Queries the marker amplitude of the selected marker.
CALCulate:IQVTime subgroup	**RF I&Q versus Time measurement**
CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:DELTa:X? (on page 439)	Returns the delta marker time for the selected marker.
CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:DELTa:Y? (on page 439)	Returns the delta marker amplitude for the selected marker.
CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:MAXimum (on page 440)	Moves the marker to the highest peak on the trace.
CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (on page 441)	Moves the marker to the next peak higher in amplitude.
CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:PEAK:LEFT (on page 441)	Moves the marker to the next peak to the left on the trace.
CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:PEAK:LOWer (on page 442)	Moves the marker to the next peak lower in amplitude.
CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (on page 442)	Moves the marker to the next peak to the right on the trace.
CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:TRACe (on page 443)	Sets or queries the trace (I or Q) to place the marker.
CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:X (on page 443)	Sets or queries the horizontal position of the marker.
CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:Y? (on page 444)	Queries the marker amplitude of the selected marker.
CALCulate:MCPower subgroup	**MCPR measurement**
CALCulate:MCPower:MARKer<x>:DELTa:X? (on page 464)	Returns the delta marker frequency for the selected marker.
CALCulate:MCPower:MARKer<x>:DELTa:Y? (on page 464)	Returns the delta marker amplitude for the selected marker.
CALCulate:MCPower:MARKer<x>:MAXimum (on page 465)	Moves the marker to the highest peak on the trace.
CALCulate:MCPower:MARKer<x>:PEAK:LEFT (on page 465)	Moves the marker to the next peak to the left on the trace.
CALCulate:MCPower:MARKer<x>:PEAK:RIGHt (on page 466)	Moves the marker to the next peak to the right on the trace.
CALCulate:MCPower:MARKer<x>:X (on page 466)	Sets or queries the horizontal position of the marker.

Command	Description
CALCulate:MCPower:MARKer<x>:Y? (on page 466)	Queries the marker amplitude of the selected marker.
CALCulate:MERRor:Subgroup	**Magnitude error versus Time measurement**
CALCulate:MERRor[:MEASview<y>]:MARKer<x>:DELTa:X? (on page 467)	Returns the delta marker amplitude for the selected marker.
CALCulate:MERRor[:MEASview<y>]:MARKer<x>:DELTa:Y? (on page 468)	Returns the delta marker amplitude for the selected marker.
CALCulate:MERRor[:MEASview<y>]:MARKer<x>:MAXimum (on page 468)	Moves the marker to the highest peak on the trace.
CALCulate:MERRor[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (on page 469)	Moves the marker to the next peak higher in amplitude.
CALCulate:MERRor[:MEASview<y>]:MARKer<x>:PEAK:LEFT (on page 469)	Moves the selected marker to the next peak to the left on the trace.
CALCulate:MERRor[:MEASview<y>]:MARKer<x>:PEAK:LOWer (on page 470)	Moves the selected marker to the next peak lower in amplitude.
CALCulate:MERRor[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (on page 471)	Moves the selected marker to the next peak to the right on the trace.
CALCulate:MERRor[:MEASview<y>]:MARKer<x>:X (on page 471)	Sets or queries the horizontal position of the marker.
CALCulate:MERRor[:MEASview<y>]:MARKer<x>:Y? (on page 472)	Queries the marker amplitude of the selected marker.
CALCulate:OBWidth subgroup	**Occupied Bandwidth measurement**
CALCulate:OBWidth:MARKer<x>:DELTa:X? (on page 494)	Returns the delta marker frequency for the selected marker.
CALCulate:OBWidth:MARKer<x>:DELTa:Y? (on page 494)	Returns the delta marker amplitude for the selected marker.
CALCulate:OBWidth:MARKer<x>:MAXimum (on page 495)	Moves the marker to the highest peak on the trace.
CALCulate:OBWidth:MARKer<x>:PEAK:HIGHer (on page 495)	Moves the marker to the next peak higher in amplitude.
CALCulate:OBWidth:MARKer<x>:PEAK:LEFT (on page 496)	Moves the marker to the next peak to the left on the trace.
CALCulate:OBWidth:MARKer<x>:PEAK:LOWer (on page 496)	Moves the marker to the next peak lower in amplitude.
CALCulate:OBWidth:MARKer<x>:PEAK:RIGHt (on page 497)	Moves the marker to the next peak to the right on the trace.

Command	Description
CALCulate:OBWidth:MARKer<x>[:SET]:CENTer (on page 497)	Sets the center frequency to the value at the marker position.
CALCulate:OBWidth:MARKer<x>:X (on page 497)	Sets or queries the horizontal position of the marker.
CALCulate:OBWidth:MARKer<x>:Y? (on page 498)	Queries the marker amplitude of the selected marker.
CALCulate:OFDM subgroup	**Orthogonal frequency division multiplexing measurement**
CALCulate:OFDM:CONSte:MARKer<x>:FREQuency (on page 498)	Sets or queries the frequency position of the selected marker.
CALCulate:OFDM:CONSte:MARKer<x>:MAGNitude? (on page 499)	Queries the marker magnitude of the selected marker.
CALCulate:OFDM:CONSte:MARKer<x>:PHASe? (on page 499)	Queries the phase readout of the selected marker.
CALCulate:OFDM:CONSte:MARKer<x>:TIME (on page 500)	Sets or queries the X position of the selected marker.
CALCulate:OFDM:CONSte:MARKer<x>:TYPE? (on page 500)	Queries the data type of the selected marker.
CALCulate:OFDM:CONSte:MARKer<x>:VALue? (on page 501)	Queries the value of the specified marker.
CALCulate:OFDM:CRESponse:MARKer<x>:DELTa:X? (on page 501)	Returns the delta marker frequency for the selected marker.
CALCulate:OFDM:CRESponse:MARKer<x>:DELTa:Y? (on page 502)	Returns the delta marker frequency for the selected marker.
CALCulate:OFDM:CRESponse:MARKer<x>:MAXimum (on page 502)	Moves the selected marker to the highest peak.
CALCulate:OFDM:CRESponse:MARKer<x>:PEAK:HIGHer (on page 502)	Moves the selected marker to the next peak higher in amplitude.
CALCulate:OFDM:CRESponse:MARKer<x>:PEAK:LEFT (on page 503)	Moves the selected marker to the next peak to the left.
CALCulate:OFDM:CRESponse:MARKer<x>:PEAK:LOWer (on page 503)	Moves the selected marker to the next peak lower in amplitude.
CALCulate:OFDM:CRESponse:MARKer<x>:PEAK:RIGHt (on page 504)	Moves the selected marker to the next peak to the right.
CALCulate:OFDM:CRESponse:MARKer<x>:X (on page 504)	Sets or queries the horizontal position of the selected marker.
CALCulate:OFDM:CRESponse:MARKer<x>:Y? (on page 505)	Returns the amplitude of the selected marker.

Command	Description
CALCulate:OFDM:FLATness:MARKer<x>:MAXimum (on page 505)	Positions the specified marker at the maximum point on the trace.
CALCulate:OFDM:FLATness:MARKer<x>:PEAK:HIGHer (on page 505)	Moves the specified marker to the next peak higher in amplitude on the trace.
CALCulate:OFDM:FLATness:MARKer<x>:PEAK:LEFT (on page 506)	Moves the specified marker to the next peak to the left on the trace.
CALCulate:OFDM:FLATness:MARKer<x>:PEAK:LOWer (on page 506)	Moves the marker to the next peak lower in amplitude on the trace.
CALCulate:OFDM:FLATness:MARKer<x>:PEAK:RIGHt (on page 507)	Moves the marker to the next peak to the right on the trace.
CALCulate:OFDM:FLATness:MARKer<x>:X (on page 507)	Sets or queries the horizontal value at the selected marker position on the points trace.
CALCulate:OFDM:FLATness:MARKer<x>:Y? (on page 508)	Returns the value of the amplitude (vertical position) at the selected marker position.
CALCulate:OFDM:TABLe:MARKer<x>:FREQuency (on page 508)	Sets or queries the frequency position of the selected marker.
CALCulate:OFDM:TABLe:MARKer<x>:TIMe (on page 508)	Sets or queries the marker time.
CALCulate:OFDM:TABLe:MARKer<x>:VALue? (on page 509)	Queries the value readout of the selected marker.
CALCulate:PERRor subgroup (Option 21 only)	**Phase error versus Time measurement**
CALCulate:PERRor[:MEASview<y>]:MARKer<x>:DELTa:X? (on page 525)	Returns the delta marker amplitude for the selected marker in the Phase error versus Time measurement.
CALCulate:PERRor[:MEASview<y>]:MARKer<x>:DELTa:Y? (on page 526)	Returns the delta marker phase for the selected marker.
CALCulate:PERRor[:MEASview<y>]:MARKer<x>:MAXimum (on page 527)	Moves the marker to the highest peak on the trace.
CALCulate:PERRor[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (on page 527)	Moves the marker to the next peak higher in amplitude.
CALCulate:PERRor[:MEASview<y>]:MARKer<x>:PEAK:LEFT (on page 528)	Moves the marker to the next peak to the left on the trace.
CALCulate:PERRor[:MEASview<y>]:MARKer<x>:PEAK:LOWer (on page 528)	Moves the marker to the next peak lower in amplitude.
CALCulate:PERRor[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (on page 529)	Moves the marker to the next peak to the right on the trace.

Command	Description
CALCulate:PERRor[:MEASview<y>]:MARKer<x>:X (on page 530)	Sets or queries the horizontal position of the marker.
CALCulate:PERRor[:MEASview<y>]:MARKer<x>:Y? (on page 530)	Queries the marker amplitude of the selected marker.
CALCulate:PHVTime subgroup	**Phase versus Time measurement**
CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:DELTa:X? (on page 531)	Returns the delta marker time for the selected marker.
CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:DELTa:Y? (on page 531)	Returns the delta marker phase for the selected marker.
CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:MAXimum (on page 532)	Moves the marker to the highest peak on the trace.
CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (on page 532)	Moves the marker to the next peak higher in amplitude.
CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:PEAK:LEFT (on page 533)	Moves the marker to the next peak to the left on the trace.
CALCulate:PHVTime[:MEASview<y>:MARKer<x>:PEAK:LOWer (on page 533)	Moves the marker to the next peak lower in amplitude.
CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (on page 534)	Moves the marker to the next peak to the right on the trace.
CALCulate:PHVTime[:MEASview<y>]:MARKer<x> :TRACe	Sets or queries the trace on which the specified marker is placed in the Phase versus Time measurement.
CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:X (on page 535)	Sets or queries the horizontal position of the marker.
CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:Y? (on page 536)	Queries the marker amplitude of the selected marker.
CALCulate:PULSe subgroup	**Pulsed RF measurements**
CALCulate:PULSe:STATistics[:MEASview<y>]:FFT:INDicator:X (on page 543)	Sets or queries the frequency in Hz for the X cursor.
CALCulate:PULSe:STATistics[:MEASview<y>]:FFT:INDicator:Y? (on page 543)	Returns the frequency level in dB for the Y cursor.
CALCulate:PULSe:STATistics[:MEASview<y>]:HISTogram:INDicator:X (on page 544)	Sets or queries the histogram indicator X value.
CALCulate:PULSe:STATistics[:MEASview<y>]:HISTogram:INDicator:Y? (on page 544)	Queries the histogram indicator Yvalue.
CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:DELTa:X? (on page 545)	Returns the delta marker frequency for the specified marker.

Command	Description
CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:DELTa:Y? (on page 545)	Returns the delta marker amplitude for the specified marker.
CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:MAXimum (on page 546)	Moves the marker to the highest peak on the trace.
CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (on page 546)	Moves the marker to the next peak higher in amplitude.
CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:PEAK:LEFT (on page 547)	Moves the marker to the next peak to the left on the statistics trace.
CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:PEAK:LOWer (on page 548)	Moves the marker to the next peak lower in amplitude.
CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (on page 548)	Moves the marker to the next peak to the right on the statistics trace.
CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:X (on page 549)	Sets or queries the horizontal position of the marker.
CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:Y? (on page 549)	Queries the marker amplitude of the selected marker.
CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:DELTa:X? (on page 550)	Returns the delta marker time for the selected marker.
CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:DELTa:Y? (on page 550)	Returns the delta marker amplitude for the selected marker.
CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:MAXimum (on page 551)	Moves the marker to the highest peak on the pulse trace.
CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (on page 551)	Moves the marker to the next peak higher in amplitude.
CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:PEAK:LEFT (on page 552)	Moves the marker to the next peak to the left on the pulse trace.
CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:PEAK:LOWer (on page 552)	Moves the marker to the next peak lower in amplitude.
CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (on page 553)	Moves the marker to the next peak to the right on the pulse trace.
CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:X (on page 553)	Sets or queries the horizontal position of the marker.
CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:Y? (on page 554)	Queries the marker amplitude of the selected marker.
CALCulate:SEARch subgroup	**Spectrogram measurement**

Command	Description
CALCulate:SEARch:LIMit:FAIL? (on page 555)	Queries whether the waveform cuts across the limit in the search operation.
CALCulate:SEARch:LIMit:MATCh:BEEP[:STATe] (on page 555)	Sets or queries whether or not to emit a beep on match during run or replay in the search operation.
CALCulate:SEARch:LIMit:MATCh:SACQuire[:STATe] (on page 555)	Sets or queries whether or not to stop acquiring data on match during run or replay in the search operation.
CALCulate:SEARch:LIMit:MATCh:SDATa[:STATe] (on page 556)	Sets or queries whether or not to save automatically (AutoSave) acquisition data on match during run in the search operation.
CALCulate:SEARch:LIMit:MATCh:SPICture[:STATe] (on page 556)	Sets or queries whether or not to save automatically (AutoSave) the whole screen on match during run in the search operation.
CALCulate:SEARch:LIMit:MATCh:STRace[:STATe] (on page 557)	Sets or queries whether or not to save automatically (AutoSave) the spectrum trace on match during run in the search operation.
CALCulate:SEARch:LIMit:OPERation (on page 558)	Sets or queries the search condition parameter.
CALCulate:SEARch:LIMit:OPERation:FEED (on page 559)	Sets or queries the data flow to be fed in the search operation.
CALCulate:SEARch:LIMit:OPERation:MASK:LOAD (on page 559)	Loads the specified limit mask file for the search operation.
CALCulate:SEARch:LIMit:OPERation:MASK:STORE (on page 560)	Saves the limit mask to a specified file.
CALCulate:[:SEARch|:TEST]:LIMit[:STATe]:MATCh:FILE:LOCation (on page 564)	Sets or queries the limit value (in dB) in the search operation.
CALCulate:SEARch:LIMit:REPort:DATA? (on page 561)	Queries the frequency range(s) that satisfy the search condition.
CALCulate:SEARch:LIMit:REPort:POINts? (on page 561)	Queries the number of frequency range(s) that satisfy the search condition.
CALCulate:SEARch:LIMit:STATe (on page 562)	Sets or queries the search function (enabled or disabled).
CALCulate:SEM subgroup	**Spectral Emissions Mask measurement**
CALCulate:SEM:MARKer<x>:DELTa:X? (on page 572)	Returns the delta marker frequency for the selected marker.
CALCulate:SEM:MARKer<x>:DELTa:Y? (on page 573)	Returns the delta marker amplitude for the selected marker.
CALCulate:SEM:MARKer<x>:MAXimum (on page 573)	Moves the selected marker to the highest peak.
CALCulate:SEM:MARKer<x>:PEAK:HIGHer (on page 573)	Moves the selected marker to the next peak higher in amplitude.

Command	Description
CALCulate:SEM:MARKer<x>:PEAK:LEFT (on page 574)	Moves the selected marker to the next peak to the left.
CALCulate:SEM:MARKer<x>:PEAK:LOWer (on page 574)	Moves the selected marker to the next peak lower in amplitude.
CALCulate:SEM:MARKer<x>:PEAK:RIGHt (on page 575)	Moves the selected marker to the next peak to the right.
CALCulate:SEM:MARKer<x>[:SET]:CENTer (on page 575)	Moves the specified marker to the center frequency.
CALCulate:SEM:MARKer<x>:X (on page 575)	Sets or queries the horizontal position of the selected marker.
CALCulate:SEM:MARKer<x>:Y? (on page 576)	Queries the marker amplitude of the selected marker.
[CALCulate]:SGRam subgroup	**Spectrogram measurements**
CALCulate:SGRam[:MEASview<y>]:MARKer<x>:DELTa:X:FREQuency? (on page 576)	Returns the delta marker frequency for the selected marker.
CALCulate:SGRam[:MEASview<y>]:MARKer<x>:DELTa:X[:TIME]? (on page 577)	Returns the delta marker time for the selected marker.
CALCulate:SGRam[:MEASview<y>]:MARKer<x>:DELTa:Y? (on page 577)	Returns the delta marker amplitude for the selected marker.
CALCulate:SGRam[:MEASview<y>]:MARKer<x>:MAXimum (on page 578)	Moves the marker to the highest peak on the trace.
CALCulate:SGRam[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (on page 578)	Moves the marker to the next peak higher in amplitude.
CALCulate:SGRam[:MEASview<y>]:MARKer<x>:PEAK:LEFT (on page 579)	Moves the marker to the next peak to the left on the trace.
CALCulate:SGRam[:MEASview<y>]:MARKer<x>:PEAK:LOWer (on page 579)	Moves the marker to the next peak lower in amplitude.
CALCulate:SGRam[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (on page 580)	Moves the marker to the next peak to the right on the trace.
CALCulate:SGRam[:MEASview<y>]:MARKer<x>[:SET]:CENTer (on page 581)	Sets the center frequency to the marker frequency.
CALCulate:SGRam[:MEASview<y>]:MARKer<x>:X:FREQuency (on page 581)	Sets or queries the marker frequency.
CALCulate:SGRam[:MEASview<y>]:MARKer<x>:X[:TIME] (on page 582)	Sets or queries the marker time.
CALCulate:SGRam[:MEASview<y>]:MARKer<x>:Y? (on page 582)	Queries the marker amplitude of the selected marker.

Command	Description
CALCulate:SPECtrum subgroup	**Spectrogram measurement**
CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:DELTa:X? (on page 583)	Returns the delta marker frequency for the selected marker.
CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:DELTa:Y? (on page 583)	Returns the delta marker amplitude for the selected marker.
CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:MAXimum (on page 584)	Moves the marker to the highest peak on the trace.
CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (on page 584)	Moves the marker to the next peak higher in amplitude.
CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:PEAK:LEFT (on page 585)	Moves the marker to the next peak to the left on the trace.
CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:PEAK:LOWer (on page 585)	Moves the marker to the next peak lower in amplitude.
CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (on page 586)	Moves the marker to the next peak to the right on the trace.
CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>[: SET]:CENTer (on page 588)	Sets the center frequency to the marker frequency.
CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:TRACe (on page 589)	Sets or queries the trace on which the marker is placed.
CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:X (on page 589)	Sets or queries the horizontal position of the marker.
CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:Y? (on page 590)	Queries the marker amplitude of the selected marker.
CALCulate:SPURious subgroup	**Spurious measurement**
CALCulate:SPURious:MARKer<x>:DELTa:X? (on page 591)	Returns the delta marker frequency for the selected marker.
CALCulate:SPURious:MARKer<x>:DELTa:Y? (on page 591)	Returns the delta marker amplitude for the selected marker.
CALCulate:SPURious:MARKer<x>:MAXimum (on page 592)	Moves the marker to the highest peak on the trace.
CALCulate:SPURious:MARKer<x>:PEAK:HIGHer (on page 592)	Moves the marker to the next peak higher in amplitude.
CALCulate:SPURious:MARKer<x>:PEAK:LEFT (on page 592)	Moves the marker to the next peak to the left on the trace.
CALCulate:SPURious:MARKer<x>:PEAK:LOWer (on page 593)	Moves the marker to the next peak lower in amplitude.

Command	Description
CALCulate:SPURious:MARKer<x>:PEAK:RIGHt (on page 593)	Moves the marker to the next peak to the right on the trace.
CALCulate:SPURious:MARKer<x>[:SET]:CENTer (on page 593)	Sets the center frequency to the marker frequency.
CALCulate:SPURious:MARKer<x>:X (on page 594)	Sets or queries the horizontal position of the marker.
CALCulate:SPURious:MARKer<x>:Y? (on page 594)	Queries the marker amplitude of the selected marker.
CALCulate:TDIagram subgroup (Option 21 only)	**Trellis diagram**
CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:DELTa:X[:TIME]? (on page 595)	Returns the delta marker time for the selected marker.
CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:DELTa:Y? (on page 596)	Returns the delta marker amplitude for the selected marker.
CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:MAXimum (on page 597)	Moves the marker to the highest peak on the trace.
CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (on page 597)	Moves the marker to the next peak higher in amplitude on the Trellis diagram trace.
CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:PEAK:LEFT (on page 598)	Moves the marker to the next peak to the left on the trace.
CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:PEAK:LOWer (on page 598)	Moves the marker to the next peak lower in amplitude.
CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (on page 599)	Moves the marker to the next peak to the right on the trace.
CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:X[:TIME] (on page 599)	Sets or queries the horizontal position (time) of the marker.
CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:Y? (on page 600)	Queries the vertical position of the selected marker in the Trellis diagram measurement.
CALCulate:TOVerview subgroup	**Time overview**
CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:DELTa:X? (on page 601)	Returns the delta marker time for the selected marker.
CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:DELTa:Y? (on page 601)	Returns the delta marker amplitude for the selected marker.
CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:MAXimum (on page 602)	Moves the marker to the highest peak on the trace.
CALCulate:TOVerview[:MEASview<y>[:MARKer<x>:PEAK:HIGHer (on page 602)	Moves the marker to the next peak higher in amplitude.

Command	Description
CALCulate:TOVerview[:MEASview<y>[:MARKer<x>:PEAK:LEFT (on page 603)	Moves the marker to the next peak to the left on the trace.
CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:PEAK:LOWer (on page 603)	Moves the marker to the next peak lower in amplitude.
CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (on page 604)	Moves the marker to the next peak to the right on the trace.
CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:X (on page 604)	Sets or queries the horizontal position of the marker
CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:Y? (on page 605)	Queries the marker amplitude of the selected marker.
CALCulate:WLAN subgroup	**Wireless LAN measurement**
CALCulate:WLAN:CONSte:MARKer<x>:FREQuency (on page 626)	Sets or queries the frequency value of the marker.
CALCulate:WLAN:CONSte:MARKer<x>:MAGNitude? (on page 627)	Returns the magnitude value at the indicated marker position.
CALCulate:WLAN:CONSte:MARKer<x>:PHASe? (on page 627)	Returns the phase value at the indicated marker position.
CALCulate:WLAN:CONSte:MARKer<x>:TIME (on page 628)	Sets the time offset value at the selected marker position on the points trace.
CALCulate:WLAN:CONSte:MARKer<x>:TYPE? (on page 628)	Returns the data type of the subcarrier for data at the indicated marker position.
CALCulate:WLAN:CONSte:MARKer<x>:VALue? (on page 629)	Queries the value readout at the selected marker position on the points trace.
CALCulate:WLAN:CRESponse:MARKer<x>:DELTa:X? (on page 629)	Queries the delta Subcarrier or frequency value at the selected marker position.
CALCulate:WLAN:CRESponse:MARKer<x>:DELTa:Y? (on page 630)	Queries the vertical delta value for the points trace at the selected marker position.
CALCulate:WLAN:CRESponse:MARKer<x>:MAXimum (on page 630)	Moves the marker to the highest peak on the trace.
CALCulate:WLAN:CRESponse:MARKer<x>:PEAK:HIGHer (on page 631)	Moves the marker to the next peak higher in amplitude.
CALCulate:WLAN:CRESponse:MARKer<x>:PEAK:LEFT (on page 631)	Moves the marker to the next peak on the left on the trace.
CALCulate:WLAN:CRESponse:MARKer<x>:PEAK:LOWer (on page 632)	Moves the marker to the next peak lower in amplitude.

Command	Description
CALCulate:WLAN:CRESponse:MARKer<x>:PEAK:RIGHt (on page 632)	Moves the marker to the next peak to the right on the trace.
CALCulate:WLAN:CRESponse:MARKer<x>:X (on page 633)	Sets or queries the value of the horizontal position for the selected marker.
CALCulate:WLAN:CRESponse:MARKer<x>:Y? (on page 633)	Returns the value of the amplitude at the selected marker position.
CALCulate:WLAN:EVM:MARKer<x>:AVERage:SUBCarrier? (on page 634)	Queries the EVM value at the selected marker position on the average trace.
CALCulate:WLAN:EVM:MARKer<x>:AVERage:SYMBol? (on page 634)	Queries the EVM value at the selected marker position on the average trace.
CALCulate:WLAN:EVM:MARKer<x>:DELTa:SUBCarrier? (on page 635)	Queries the delta Subcarrier or frequency value at the selected marker position.
CALCulate:WLAN:EVM:MARKer<x>:DELTa:SYMBol? (on page 635)	Queries the delta time value in seconds or symbols at the selected marker position.
CALCulate:WLAN:EVM:MARKer<x>:DELTa:Y? (on page 636)	Queries the vertical delta value for the points trace at the selected marker position.
CALCulate:WLAN:EVM:MARKer<x>:FREQuency (on page 636)	Sets or queries the frequency value of the marker.
CALCulate:WLAN:EVM:MARKer<x>:TIME (on page 637)	Sets the vertical marker value of the points trace.
CALCulate:WLAN:EVM:MARKer<x>:VALue? (on page 638)	Queries the EVM value at the selected marker position.
CALCulate:WLAN:FLATness:MARKer<x>:MAXimum (on page 638)	Positions the specified marker at the maximum point on the trace.
CALCulate:WLAN:FLATness:MARKer<x>:PEAK:HIGHer (on page 639)	Moves the specified marker to the next peak higher in amplitude.
CALCulate:WLAN:FLATness:MARKer<x>:PEAK:LEFT (on page 639)	Moves the specified marker to the next peak to the left.
CALCulate:WLAN:FLATness:MARKer<x>:PEAK:LOWer (on page 640)	Moves the marker to the next peak lower in amplitude.
CALCulate:WLAN:FLATness:MARKer<x>:PEAK:RIGHt (on page 640)	Moves the marker to the next peak to the right.
CALCulate:WLAN:FLATness:MARKer<x>:X (on page 641)	Sets or queries the horizontal position for the selected marker.
CALCulate:WLAN:FLATness:MARKer<x>:Y? (on page 641)	Returns the value of the amplitude (vertical position) at the selected marker position.

Command	Description
CALCulate:WLAN:MERRor:MARKer<x>:AVERage:SUBCarrier? (on page 642)	Queries the value readout at the selected marker position on the average trace.
CALCulate:WLAN:MERRor:MARKer<x>:AVERage:SYMBol? (on page 642)	Queries the value readout at the selected marker position on the average trace.
CALCulate:WLAN:MERRor:MARKer<x>:DELTa:SUBCarrier? (on page 643)	Queries the delta Subcarrier or frequency value at the selected marker position.
CALCulate:WLAN:MERRor:MARKer<x>:DELTa:SYMBol? (on page 643)	Queries the delta time value in seconds or symbols at the selected marker position.
CALCulate:WLAN:MERRor:MARKer<x>:DELTa:Y? (on page 644)	Queries the vertical delta value for the points trace at the selected marker position.
CALCulate:WLAN:MERRor:MARKer<x>:FREQuency (on page 645)	Sets or queries the frequency value of the marker.
CALCulate:WLAN:MERRor:MARKer<x>:TIME (on page 645)	Sets the time offset value at the selected marker position.
CALCulate:WLAN:MERRor:MARKer<x>:VALue? (on page 646)	Queries the value readout at the selected marker position.
CALCulate:WLAN:PERRor:MARKer<x>:AVERage:SUBCarrier? (on page 647)	Queries the Phase Error value at the selected marker position on the average trace.
CALCulate:WLAN:PERRor:MARKer<x>:AVERage:SYMBol? (on page 647)	Queries the Phase Error value at the selected marker position on the average trace.
CALCulate:WLAN:PERRor:MARKer<x>:DELTa:SUBCarrier? (on page 648)	Queries the delta Subcarrier or frequency value at the selected marker position.
CALCulate:WLAN:PERRor:MARKer<x>:DELTa:SYMBol? (on page 648)	Queries the delta time value in seconds or symbols at the selected marker position.
CALCulate:WLAN:PERRor:MARKer<x>:DELTa:Y? (on page 649)	Queries the vertical delta value for the points trace at the selected marker position.
CALCulate:WLAN:PERRor:MARKer<x>:FREQuency (on page 649)	Sets or queries the frequency value at the selected marker position.
CALCulate:WLAN:PERRor:MARKer<x>:TIME (on page 650)	Sets the time value (offset) at the selected marker position.
CALCulate:WLAN:PERRor:MARKer<x>:VALue? (on page 651)	Queries the Phase Error value at the selected marker position.
CALCulate:WLAN:PVTime:MARKer<x>:DELTa:X? (on page 651)	Returns the delta time value at the selected marker position.
CALCulate:WLAN:PVTime:MARKer<x>:DELTa:Y? (on page 652)	Queries the vertical delta value for the points trace at the selected marker position.

Command	Description
CALCulate:WLAN:PVTime:MARKer<x>:MAXimum (on page 652)	Positions the specified marker at the maximum point.
CALCulate:WLAN:PVTime:MARKer<x>:PEAK:HIGHer (on page 653)	Moves the specified marker to the next peak higher in amplitude.
CALCulate:WLAN:PVTime:MARKer<x>:PEAK:LEFT (on page 653)	Moves the specified marker to the next peak on the left.
CALCulate:WLAN:PVTime:MARKer<x>:PEAK:LOWer (on page 654)	Moves the marker to the next peak lower in amplitude.
CALCulate:WLAN:PVTime:MARKer<x>:PEAK:RIGHt (on page 654)	Moves the marker to the next peak to the right.
CALCulate:WLAN:PVTime:MARKer<x>:X (on page 654)	Sets or queries the value of the horizontal position for the selected marker
CALCulate:WLAN:PVTime:MARKer<x>:Y? (on page 655)	Returns the amplitude at the selected marker position.
CALCulate:WLAN:TABLe:MARKer<x>:FREQuency (on page 655)	Assigns or queries the frequency value of the marker.
CALCulate:WLAN:TABLe:MARKer<x>:TIME (on page 656)	Sets or queries the vertical value at the selected marker position.
CALCulate:WLAN:TABLe:MARKer<x>:VALue? (on page 657)	Queries the decoded symbol value at the selected marker position.
CALCulate:LTE subgroup	
CALCulate:LTE:ACLR:MARKer:<x>PEAK:HIGHer (on page 444)	Moves the specified marker to the next peak higher in amplitude on the trace.
CALCulate:LTE:ACLR:MARKer<x>:DELTa:X? (on page 445)	Returns the frequency of the specified delta marker on the trace.
CALCulate:LTE:ACLR:MARKer<x>:DELTa:Y? (on page 445)	Returns the amplitude for the specified delta marker on the trace.
CALCulate:LTE:ACLR:MARKer<x>:MAXimum (on page 446)	Positions the specified marker at the highest peak on the trace.
CALCulate:LTE:ACLR:MARKer<x>:PEAK:LEFT (on page 446)	Moves the selected marker to the next peak to the left on the trace.
CALCulate:LTE:ACLR:MARKer<x>:PEAK:LOWer (on page 446)	Moves the selected marker to the next peak lower in amplitude on the trace.
CALCulate:LTE:ACLR:MARKer<x>:PEAK:RIGHt (on page 447)	Moves the selected marker to the next peak to the right on the trace.

Command	Description
CALCulate:LTE:ACLR:MARKer<x>:SET:CENTer (on page 447)	Moves the specified marker to the center frequency.
CALCulate:LTE:ACLR:MARKer<x>:X (on page 448)	Sets or queries the horizontal position of the selected marker on the trace.
CALCulate:LTE:ACLR:MARKer<x>:Y? (on page 448)	Queries the amplitude of the selected marker on the trace.
CALCulate:LTE:CHSPectrum:MARKer<x>:DELTa:X? (on page 449)	Returns the frequency of the specified delta marker on the trace.
CALCulate:LTE:CHSPectrum:MARKer<x>:DELTa:Y? (on page 449)	Returns the amplitude for the specified delta marker on the trace.
CALCulate:LTE:CHSPectrum:MARKer<x>:MAXimum (on page 449)	Positions the marker at the highest peak on the trace.
CALCulate:LTE:CHSPectrum:MARKer<x>:PEAK:LEFT (on page 450)	Moves the selected marker to the next peak to the left on the trace.
CALCulate:LTE:CHSPectrum:MARKer<x>:PEAK:RIGHt (on page 450)	Moves the selected marker to the next peak to the right on the trace.
CALCulate:LTE:CHSPectrum:MARKer<x>:X (on page 451)	Sets or queries the horizontal position of the selected marker on the trace.
CALCulate:LTE:CHSPectrum:MARKer<x>:Y? (on page 451)	Queries the amplitude of the selected marker on the trace.
CALCulate:LTE:CONSte:MARKer<x>:FREQuency (on page 452)	Sets or returns the subcarrier number of the LTE symbol.
CALCulate:LTE:CONSte:MARKer<x>:MAGNitude? (on page 452)	Queries the marker magnitude of the selected marker.
CALCulate:LTE:CONSte:MARKer<x>:PHASe? (on page 453)	Queries the phase readout of the selected marker.
CALCulate:LTE:CONSte:MARKer<x>:TIME (on page 453)	Sets or queries the X position (Symbol Value) of the selected marker.
CALCulate:LTE:CONSte:MARKer<x>:TYPE? (on page 454)	Queries the data type of the selected marker as PSS or SSS.
CALCulate:LTE:PVTime:MARKer:<x>PEAK:HIGHer (on page 454)	Moves the specified marker to the next peak higher in amplitude on the trace.
CALCulate:LTE:PVTime:MARKer<x>:DELTa:X? (on page 454)	Returns the frequency of the specified delta marker on the trace.
CALCulate:LTE:PVTime:MARKer<x>:DELTa:Y? (on page 455)	Returns the amplitude for the specified delta marker on the trace.

Command	Description
CALCulate:LTE:PVTime:MARKer<x>:MAXimum (on page 455)	Positions the specified marker at the highest peak on the trace.
CALCulate:LTE:PVTime:MARKer<x>:PEAK:LEFT (on page 456)	Moves the selected marker to the next peak to the left on the trace.
CALCulate:LTE:PVTime:MARKer<x>:PEAK:LOWer (on page 456)	Moves the selected marker to the next peak lower in amplitude on the trace.
CALCulate:LTE:PVTime:MARKer<x>:PEAK:RIGHt (on page 457)	Moves the selected marker to the next peak to the right on the trace.
CALCulate:LTE:PVTime:MARKer<x>:X (on page 457)	Sets or queries the horizontal position of the selected marker on the trace.
CALCulate:LTE:PVTime:MARKer<x>:Y? (on page 458)	Queries the amplitude of the selected marker on the trace.
CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:HIGHer? (on page 490)	Places the selected marker at the next highest peak on the trace.
CALCulate:PULSe:CUMulative:HISTogram[:MEASview<y>]:INDicator:X (on page 536)	Sets or queries the horizontal position of the indicator on the trace.
CALCulate:PULSe:CUMulative:HISTogram[:MEASview<y>]:INDicator:Y? (on page 537)	Queries the vertical position value of the indicator on the trace.
CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:FVTime:DELTa:X? (on page 537)	Returns the delta marker time for the selected marker.
CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:FVTime:DELTa:Y? (on page 538)	Returns the delta marker amplitude for the selected marker.
CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:FVTime:MAXimum (on page 538)	Moves the selected marker to the highest peak on the trace.
CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:FVTime:X (on page 539)	Sets or queries the horizontal position of the selected marker.
CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:FVTime:Y (on page 539)	Sets or queries the marker amplitude of the selected marker.
CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:FVTime:DELTa:X? (on page 537)	Returns the delta marker time for the selected marker.

Marker mnemonics

Up to five markers can be used. In commands, these are named MARKer<x>,where <x> can be 0, 1, 2, 3, or 4 as shown in the following table.

Marker mnemonics

Mnemonic	Description
MARKer0	Reference marker (MR)
MARKer1	Marker 1 (M1)
MARKer2	Marker 2 (M2)
MARKer3	Marker 3 (M3)
MARKer4	Marker 4 (M4)

NOTE: *If you omit the numeric suffix, the marker control defaults to Marker 1.*

Before operating the marker, you have to enable it using the CALCulate basic commands.

If you attempt to use a marker other than above in a CALCulate command, the suffix error (error code -130) will occur.

Calibration commands

Use the CALibration commands to control the external correction. See the Index for commands not listed in the following table.

Not all of these commands apply to SignalVu. See the individual command descriptions for details.

Calibration commands

Command	Description
CALibration:ABORt (on page 657)	Aborts any actions related to the alignments in progress.
CALibration:TXGain:ABORt (on page 663)	Cancels a calibration for the Transmission Gain measurement.
CALibration:TXGain[:ALL] (on page 663)	Performs a calibration using the current settings for the Transmission Gain measurement.
CALibration:TXGain:FINish? (on page 663)	Queries whether or not the calibration is completed for the Transmission Gain measurement.
CALibration:TXGain:STATe? (on page 664)	Queries the calibration state for the Transmission Gain measurement.
CALibration:AUTO (on page 658)	Sets or queries whether or not to run alignments automatically.
INPut:CORRection:EXTernal:EDIT<x>:LABel (on page 1482)	Sets or queries the name of the external loss table.
CALibration:CORRection:EXTernal:EDIT<x>:NEW (on page 658)	Creates a new external loss table.
INPut:CORRection:EXTernal:EDIT<x>:STATe (on page 1483)	Sets or queries whether to enable or disable the external loss table.
INPut:CORRection:EXTernal:GAIN[:MAGNitude] (on page 1484)	Sets or queries the external gain value.

Command	Description
CALibration:CORRection:EXTernal:PROBe:CONNect? (on page 661)	Sets or queries whether to enable or disable the external gain value.
CALibration:CORRection:EXTernal:PROBe:CONNect? (on page 661)	Queries whether the external probe is connected to the analyzer or not.
CALibration:CORRection:EXTernal:PROBe[:MAGNitude]? (on page 661)	Queries the external probe attenuation.
CALibration:CORRection:EXTernal:PROBe:STATe (on page 660)	Determines whether to correct data for the external probe attenuation.
CALibration:CORRection:EXTernal:TYPE (on page 660)	Selects data type to use when applying the external loss table.

Display commands

Use the DISPlay commands to control the display of measurement waveforms and results on the screen. See the Index for commands not listed in the following table.

Not all of these commands apply to SignalVu. See the individual command descriptions for details.

Display commands

Command	Description
DISPlay TXGain subgroup	**Transmission Gain measurements**
DISPlay:TG:MEASview:DELete TXGain (on page 978)	Removes the Transmission Gain display icon from the Selected displays filed of the Selected Displays window.
DISPlay:TG:MEASview<y>:NEW TXGain (on page 978)	Opens a new Transmission Gain measurement display window.
DISPlay:TG:MEASview<y>:SELect TXGain (on page 978)	Selects the Transmission Gain display icon in the Select Display window and queries if the display is selected or not.
DISPlay:TXGain:MARKer:SHOW:STATe (on page 985)	Displays or hides the marker readout, but not the marker itself, on the graph area.
DISPlay:TXGain[:SCALe]:AUTO (on page 985)	Automatically rescales the horizontal and vertical scale.
DISPlay:TXGain:WINDow:TRACe:GRATicule:GRID:STATe (on page 986)	Displays or hides the graticule grid in the display. Queries whether the graticule grid is hidden or showing.
DISPlay:TXGain:WINDow:TRACe:POINts:SHOW:STATe (on page 986)	Sets to mark each measurement point on the trace.
DISPlay:TXGain:X[:SCALe]:LOG:STATe (on page 987)	Sets or queries the display to show the frequency axis in a logarithmic scale.
DISPlay:TXGain:X[:SCALe]:STARt (on page 987)	Sets the Zoom Start frequency.

Command	Description
DISPlay:TXGain:X[:SCALe]:AUTO (on page 987)	Resets the scale of the horizontal axis to contain the complete trace.
DISPlay:TXGain:X[:SCALe]:STOP (on page 988)	Sets the Zoom Stop frequency.
DISPlay:TXGain:Y[:SCALe]:BOTTom (on page 989)	Sets or queries the bottom of the vertical position of the display.
DISPlay:TXGain:Y[:SCALe]:PDIVision (on page 990)	Sets or queries the vertical scale (per division) of the graph.
DISPlay:TXGain:Y[:SCALe]:POSition (on page 990)	Sets or queries the vertical position of the trace.
DISPlay:TXGain:Y[:SCALe]:TOP (on page 991)	Sets or queries the top of the vertical scale.
DISPlay:TXGain:Y[:SCALe]:AUTO (on page 989)	Automatically scales the plot vertically.
DISPlay:TXGain:Y[:SCALe] (on page 988)	Sets or queries the vertical scale value in dB.
DISPlay Noise subgroup	**Noise Figure and Gain measurements**
DISPlay:WINDow:ACTive:MEASurement? (on page 991)	Queries the active measurement views.
DISPlay Bluetooth subgroup	**Bluetooth measurements**
DISPlay:BIBEmissions:MARKer<x>:SHOW:STATe (on page 695)	Sets or queries to show or hide the readout for the selected marker.
DISPlay:BIBEmissions:PLEVel:SHOW:STATe (on page 695)	Sets or queries to show or hide the power levels
DISPlay:BIBEmissions:RESet:SCALe (on page 696)	Resets the horizontal and vertical scale to the default values.
DISPlay:BIBEmissions:WINDow:TRACe:GRATicule:GRID:STATe (on page 696)	Sets or queries the graticule hidden or showing state.
DISPlay:BIBEmissions:X[:SCALe] (on page 697)	Sets or queries the horizontal range of the graph.
DISPlay:BIBEmissions:X[:SCALe]:AUTO (on page 697)	Rescales the horizontal axis automatically to fit the waveform to the screen in the Bluetooth InBand Emission view.
DISPlay:BIBEmissions:X[:SCALe]:OFFSet (on page 698)	Sets or queries the center frequency.
DISPlay:BIBEmissions:Y[:SCALe] (on page 698)	Sets or queries the vertical range of the graph.
DISPlay:BIBEmissions:Y[:SCALe]:AUTO (on page 699)	Rescales the vertical axis automatically to fit the waveform to the screen in the Bluetooth InBand Emission view.
DISPlay:BIBEmissions:Y[:SCALe]:OFFSet (on page 699)	Sets or queries the vertical offset.
DISPlay:BLUEtooth:CONSte:TRACe:GRATicule:GRID:STATe (on page 700)	Sets or queries the display state of graticule grid.

Command	Description
DISPlay:BLUEtooth:CONSte:WINDow:TRACe:GRATicule:GRID:STATe (on page 700)	Sets or queries whether to show or hide the graticule grid on the screen in the display.
DISPlay:BLUEtooth:EDIagram:WINDow:TRACe:GRATicule:GRID:STATe (on page 701)	Sets or queries graticule grid hidden or showing state in the Bluetooth eye diagram display.
DISPlay:BLUEtooth:EDIagram:Y[:SCALe] (on page 701)	Sets or queries the vertical range.
DISPlay:BLUEtooth:EDIagram:Y[:SCALe]:AUTO (on page 702)	Sets the vertical scale automatically to fit the waveform in the Bluetooth eye diagram.
DISPlay:BLUEtooth:EDIagram:Y[:SCALe]:OFFSet (on page 702)	Sets or queries the vertical offset (center point of the vertical axis).
DISPlay:BLUEtooth:FDVTime:WINDow:TRACe:GRATicule:GRID:STATe (on page 702)	Sets or queries graticule grid hidden or showing state in the Freq Dev vs Time display.
DISPlay:BLUEtooth:FDVTime:Y[:SCALe]:AUTO (on page 703)	Rescales the vertical scale automatically to fit the waveform to the screen.
DISPlay:BLUEtooth:FDVTime:Y[:SCALe]:OFFSet (on page 703)	Sets or queries the vertical offset (center point of the vertical axis).
DISPlay:BLUEtooth:MEASview<y>:DELete (on page 704)	Deletes the specified Bluetooth analysis view.
DISPlay:BLUEtooth:MEASview<y>:NEW (on page 704)	Displays a new measurement view for Bluetooth measurements.
DISPlay:BLUEtooth:MEASview<y>:SELect (on page 705)	Selects a measurement view on the screen in the Bluetooth measurements.The query command returns the currently selected view.
DISPlay:BOBW:SELected:BANDwidth (on page 706)	Sets or queries the bandwidth to measure in the Bluetooth 20 dB BW view.
DISPlay:DPX:WINDow:TRACe:GRATicule:GRID:STATe (on page 723)	Determines whether to show or hide the graticule grid on the DPX Spectrum display..
DISPlay:NOISe:FIGure[:SCALe]:AUTO (on page 791)	Automatically scales the graph.
DISPlay:NOISe:FIGure:WINDow:TRACe:GRATicule:GRID:STATe (on page 792)	Sets or queries the graticule state to on (showing) or off (hidden).
DISPlay:NOISe:FIGure:WINDow:TRACe:LEGend:STATe (on page 792)	Sets or queries the trace legend state to on (showing) or off (hidden).
DISPlay:NOISe:FIGure:WINDow:TRACe:MEASpoints:STATe (on page 793)	Sets or queries the measurement points state to on (showing) or off (hidden).
DISPlay:NOISe:FIGure:X:AUTO (on page 793)	Automatically sets the horizontal scale of the graph.
DISPlay:NOISe:FIGure:X:OFFSet (on page 794)	Sets or queries the center frequency (offset).

Command	Description
DISPlay:NOISe:FIGure:X[:SCALe] (on page 794)	Sets or queries the horizontal scale of the display.
DISPlay:NOISe:FIGure:Y:AUTO (on page 795)	Automatically sets the vertical scale of the graph.
DISPlay:NOISe:FIGure:Y:AUTO:STATe (on page 795)	Sets the state of the automatic vertical scale and position of the graph to on or off. The query form of the command returns the current state.
DISPlay:NOISe:FIGure:Y:PDIVision (on page 796)	Sets or queries the distance between graticule lines on the display.
DISPlay:NOISe:FIGure:Y:POSition (on page 796)	Sets or queries the vertical position of the display.
DISPlay:NOISe:FIGure:Y:SCALe (on page 797)	Sets or queries the vertical scale of the display.
DISPlay:NOISe:FIGure:Y:SCALe:LINear:STATe (on page 797)	Sets or queries whether the linear units are shown (on) or not shown (off) on the display.
DISPlay:NOISe:GAIN[:SCALe]:AUTO (on page 798)	Automatically scales the graph.
DISPlay:NOISe:GAIN:WINDow:TRACe:GRATicule:GRID:STATe (on page 798)	Sets or queries the graticule state to on (showing) or off (hidden).
DISPlay:NOISe:GAIN:WINDow:TRACe:LEGend: STATe (on page 799)	Sets or queries the trace legend state to on (showing) or off (hidden).
DISPlay:NOISe:GAIN:WINDow:TRACe:MEASPoints:STATe (on page 799)	Sets or queries the measurement points state to on (showing) or off (hidden).
DISPlay:NOISe:GAIN:X:AUTO (on page 800)	Automatically sets the horizontal scale of the display.
DISPlay:NOISe:GAIN:X:OFFSet (on page 800)	Sets or queries the center frequency (offset).
DISPlay:NOISe:GAIN:X:SCALe (on page 800)	Sets or queries the horizontal scale of the display.
DISPlay:NOISe:GAIN:Y:AUTO (on page 801)	Automatically sets the vertical scale of the graph.
DISPlay:NOISe:GAIN:Y:AUTO:STATe (on page 801)	Sets or queries the state of the automatic vertical scale and position of the graph to on or off.
DISPlay:NOISe:GAIN:Y:PDIVision (on page 802)	Sets or queries the distance between graticule lines on the display.
DISPlay:NOISe:GAIN:Y:POSition (on page 803)	Sets or queries the vertical position of the display.
DISPlay:NOISe:GAIN:Y:SCALe (on page 803)	Sets or queries the vertical scale of the display.
DISPlay:NOISe:MEASview<y>:DELete (on page 804)	Deletes the selected Noise and Gain measurement view.
DISPlay:NOISe:MEASview<y>:NEW (on page 804)	Displays a new Noise and Gain measurement view.
DISPlay:NOISe:MEASview<y>:SELect (on page 805)	Selects a noise measurement view. The query command returns the currently selected view.

Command	Description
DISPlay:NOISe:POWer:LINear:STATe (on page 806)	Sets or queries whether the linear units are shown (on) or not shown (off) on the Noise Table display.
DISPlay:NOISe:TEMPerature[:SCALe]:AUTO (on page 806)	Automatically scales the graph.
DISPlay:NOISe:TEMPerature:WINDow:TRACe:GRATicule:GRID:STATe (on page 806)	Sets or queries the graticule state to on (showing) or off (hidden).
DISPlay:NOISe:TEMPerature:WINDow:TRACe:LEGend:STATe (on page 807)	Sets or queries the trace legend state to on (showing) or off (hidden).
DISPlay:NOISe:TEMPerature:WINDow:TRACe:MEASPoints:STATe (on page 808)	Sets or queries the measurement points state to on (showing) or off (hidden).
DISPlay:NOISe:TEMPerature:X:AUTO (on page 808)	Automatically sets the horizontal scale of the graph.
DISPlay:NOISe:TEMPerature:X:OFFSet (on page 809)	Sets or queries the center frequency (offset) value.
DISPlay:NOISe:TEMPerature:X[:SCALe] (on page 809)	Sets or queries the horizontal scale of the graph.
DISPlay:NOISe:TEMPerature:X:STARt (on page 810)	Sets or queries the Start Frequency value.
DISPlay:NOISe:TEMPerature:X:STOP (on page 810)	Sets or queries the Stop Frequency value.
DISPlay:NOISe:TEMPerature:Y:AUTO (on page 811)	Automatically sets the vertical scale of the graph.
DISPlay:NOISe:TEMPerature:Y:AUTO:STATe (on page 811)	Sets or queries the state of the automatic vertical scale and position of the graph to on or off.
DISPlay:NOISe:TEMPerature:Y:PDIVision (on page 811)	Sets or queries the distance between graticule lines on the display.
DISPlay:NOISe:TEMPerature:Y:POSition (on page 812)	Sets or queries the vertical position of the graph.
DISPlay:NOISe:TEMPerature:Y:SCALe (on page 813)	Sets or queries the vertical scale of the graph.
DISPlay:NOISe:UNCERTainty:RESult:GAIN? (on page 813)	Queries the gain computed uncertainty value.
DISPlay:NOISe:UNCERTainty:RESult:NFIGure? (on page 813)	Queries the noise figure computed uncertainty value.
DISPlay:NOISe:YFACtor[:SCALe]:AUTO (on page 814)	Automatically scales the Y Factor graph.
DISPlay:NOISe:YFACtor:WINDow:TRACe:GRATicule:GRID:STATe (on page 814)	Sets or queries the graticule state to on (showing) or off (hidden).
DISPlay:NOISe:YFACtor:WINDow:TRACe:LEGend:STATe (on page 815)	Sets or queries the trace legend state to on (showing) or off (hidden).

Command	Description
DISPlay:NOISe:YFACtor:WINDow:TRACe:MEASPoints:STATe (on page 815)	Sets or queries the measurement points state to on (showing) or off (hidden).
DISPlay:NOISe:YFACtor:X:AUTO (on page 816)	Automatically sets the horizontal scale.
DISPlay:NOISe:YFACtor:X:OFFSet (on page 816)	Sets or queries the center frequency (offset).
DISPlay:NOISe:YFACtor:X[:SCALe] (on page 817)	Sets or queries the horizontal scale.
DISPlay:NOISe:YFACtor:X:STARt (on page 817)	Sets or queries the Start Frequency value.
DISPlay:NOISe:YFACtor:X:STOP (on page 818)	Sets or queries the Stop Frequency value.
DISPlay:NOISe:YFACtor:Y:AUTO (on page 818)	Automatically sets the vertical scale of the graph.
DISPlay:NOISe:YFACtor:Y:AUTO:STATe (on page 819)	Sets or queries the automatic vertical scale and position of the graph setting to on or off.
DISPlay:NOISe:YFACtor:Y:PDIVision (on page 819)	Sets or queries the distance between graticule lines on the display in dB.
DISPlay:NOISe:YFACtor:Y:POSition (on page 820)	Sets or queries the vertical position.
DISPlay:NOISe:YFACtor:Y:SCALe (on page 820)	Sets or queries the vertical scale.
DISPlay:P25:CONSte:WINDow:TRACe:GRATicule:GRID:STATe (on page 855)	Sets or queries to show or hide the graticule grid on the screen.
DISPlay:P25:EDIagram:WINDow:TRACe:GRATicule:GRID:STATe (on page 855)	Sets or queries to show or hide the graticule grid on the screen.
DISPlay:P25:EDIagram:Y[:SCALe] (on page 856)	Sets or queries the vertical scale.
DISPlay:P25:EDIagram:Y[:SCALe]:AUTO (on page 857)	Sets the vertical scale automatically to fit the waveform to the screen.
DISPlay:P25:EDIagram:Y[:SCALe]:OFFSet (on page 857)	Sets or queries the vertical offset (center point of the vertical axis).
DISPlay:P25:MEASview:DELete (on page 857)	Deletes the specified P25 measurement display.
DISPlay:P25:MEASview:NEW (on page 858)	Displays a new P25 measurement view.
DISPlay:P25:MEASview:SELect (on page 859)	Selects a P25 measurement display. The query form returns the currently selected display.
DISPlay:P25:PVTime:BURSt:X[:SCALe] (on page 859)	Sets or queries the value of the scale (width) value, in seconds, when using horizontal Full Burst view.
DISPlay:P25:PVTime:BURSt:X[:SCALe]:AUTO (on page 860)	Automatically sets the starting time (position) and scale (width) values for the best display in the horizontal Full Burst view.

Command	Description
DISPlay:P25:PVTime:BURSt:X[:SCALe]:OFFSet (on page 860)	Sets or queries the starting time (position) value, in seconds, when using the horizontal Full Burst view.
DISPlay:P25:PVTime:BURSt:X[:SCALe]:RESet (on page 861)	Resets the starting time (position) and scale (width) values for the horizontal Full Burst view.
DISPlay:P25:PVTime:FALL:X[:SCALe] (on page 861)	Sets or queries the scale (width) value, in seconds, when using the horizontal Falling Edge view.
DISPlay:P25:PVTime:FALL:X[:SCALe]:AUTO (on page 862)	Automatically sets the starting time (position) and scale (width) values for the best display in the horizontal Falling Edge view.
DISPlay:P25:PVTime:FALL:X[:SCALe]:OFFSet (on page 862)	Sets or queries the value for the starting time (offset) of the graph, in seconds, when using the horizontal Falling Edge view.
DISPlay:P25:PVTime:FALL:X[:SCALe]:RESet (on page 863)	Resets the starting time (position) and scale (width) values for the horizontal Falling Edge view.
DISPlay:P25:PVTime:MARKer:SHOW:STATe (on page 863)	Shows or hides the marker readout.
DISPlay:P25:PVTime:RISE:X[:SCALe] (on page 864)	Sets or queries the scale (width) value, in seconds, for the P25 Power vs. Time display when using the horizontal Rising Edge view.
DISPlay:P25:PVTime:RISE:X[:SCALe]:AUTO (on page 864)	Sets the scale (width) value, in seconds, to automatic.
DISPlay:P25:PVTime:RISE:X[:SCALe]:OFFSet (on page 865)	Sets or queries the starting time value (offset), in seconds, for the P25 Power vs. Time display when using the horizontal Rising Edge view.
DISPlay:P25:PVTime:RISE:X[:SCALe]:RESet (on page 865)	Resets the starting time (position) and scale (width) values for the horizontal Rising Edge view.
DISPlay:P25:PVTime:WINDow:SELect:PLOT (on page 866)	Sets or queries which view to use: Full Burst, Rising Edge, or Falling Edge.
DISPlay:P25:PVTime:WINDow:TRACe:GRATicule:GRID:STATe (on page 866)	Sets or queries the graticule state to on (showing) or off (hidden).
DISPlay:P25:PVTime:Y[:SCALe] (on page 867)	Sets or queries the vertical scale value.
DISPlay:P25:PVTime:Y[:SCALe]:AUTO (on page 867)	Automatically selects the vertical scale and position values.
DISPlay:P25:PVTime:Y[:SCALe]:OFFSet (on page 868)	Sets or queries the value of the vertical offset (top edge of the vertical axis).
DISPlay:P25:PVTime:Y[:SCALe]:PDIVision (on page 868)	Sets or queries the value of the vertical scale for the P25 Power vs. Time display.
DISPlay:P25:RADix (on page 869)	Set or queries the symbol radix for the P25 Symbol Table display.
DISPlay:TOVerview[:MEASview<y>]:WINDow:NAVigator:STATe (on page 979)	Sets or queries whether the navigator view of the Time Overview display is on or off.

Command	Description
DISPlay:TOVerview[:MEASview<y>]:WINDow:TIME:MODE (on page 980)	Sets or queries the type of time analysis to be performed.
DISPlay:TOVerview[:MEASview<y>]:WINDow:TRACe:LEGend:STATe (on page 981)	Sets or queries the trace legend state to on (showing) or off (hidden).
DISPlay basic command subgroup	**General window control**
DISPlay:WINDow:COLor:SCHeme (on page 994)	Sets or queries the color scheme for traces and background.
DISPlay:WINDow:OPTimized:MEASurement? (on page 994)	Queries the measurement views that are optimized.
DISPlay:ACPower subgroup	**Channel power and ACPR measurement**
DISPlay:ACPower:MARKer:SHOW:STATe (on page 665)	Determines whether to show the readout for the selected marker.
DISPlay:ACPower:PLEVel:SHOW:STATe (on page 666)	Determines whether to show the power levels.
DISPlay:ACPower:RESet:SCALe (on page 666)	Resets the horizontal and vertical scale to the default values.
DISPlay:ACPower:WINDow:TRACe:GRATicule:GRID:STATe (on page 666)	Determines whether to show the graticule grid on screen.
DISPlay:ACPower:X[:SCALe] (on page 667)	Sets or queries the horizontal range.
DISPlay:ACPower:X[:SCALe]:AUTO (on page 667)	Rescales the horizontal axis automatically.
DISPlay:ACPower:X[:SCALe]:OFFSet (on page 668)	Sets or queries the minimum horizontal value (left edge).
DISPlay:ACPower:Y[:SCALe] (on page 668)	Sets or queries the vertical range.
DISPlay:ACPower:Y[:SCALe]:AUTO (on page 669)	Rescales the vertical axis automatically.
DISPlay:ACPower:Y[:SCALe]:OFFSet (on page 669)	Sets or queries the vertical offset.
DISPlay:ADEMod subgroup (Option 21 only)	**General purpose analog demodulation measurements**
DISPlay:ADEMod:MEASview:DELete (on page 668)	Deletes the measurement view.
DISPlay:ADEMod:MEASview:NEW (on page 668)	Displays a new measurement view.
DISPlay:ADEMod:MEASview:SELect (on page 671)	Sets or queries the measurement view.
DISPlay:{AM|FM|PM} subgroup (Option 21 only)	**AM/FM/PM measurement**
DISPlay:{AM|FM|PM}:MARKer:SHOW:STATe (on page 671)	Determines whether to show the readout for the selected marker.
DISPlay:{AM|FM|PM}:WINDow:TRACe:GRATicule:GRID:STATe (on page 672)	Determines whether to show the graticule grid on screen.

Command	Description
DISPlay:{AM|FM|PM}:X:RSCale (on page 672)	Rescales the horizontal axis automatically.
DISPlay:{AM|FM|PM}:X[:SCALe]:AUTO (on page 673)	Sets the horizontal scale automatically.
DISPlay:{AM|FM|PM}:X[:SCALe]:FULL (on page 673)	Sets or queries the horizontal scale.
DISPlay:{AM|FM|PM}:X[:SCALe]:OFFSet (on page 673)	Sets or queries the minimum horizontal value (left edge).
DISPlay:{AM|FM|PM}:Y:RSCale (on page 674)	Rescales the vertical axis automatically.
DISPlay:{AM|FM|PM}:Y[:SCALe] (on page 674)	Sets or queries the vertical scale.
DISPlay:{AM|FM|PM}:Y[:SCALe]:OFFSet (on page 675)	Sets or queries the vertical offset.
DISPlay:AUDio subgroup	**Audio measurements**
DISPlay:AUDio:MEASview<y>:DELete (on page 676)	Deletes the specified audio analysis view.
DISPlay:AUDio:MEASview<y>:NEW (on page 676)	Displays a new audio analysis view.
DISPlay:AUDio:MEASview<y>:SELect (on page 677)	Selects an audio analysis view on the screen.
DISPlay:AUDio:SPECtrum:FREQuency:[SCALe]:STARt (on page 678)	Sets or queries the start frequency (left edge) of the audio spectrum graph.
DISPlay:AUDio:SPECtrum:FREQuency:[SCALe]: STOP (on page 678)	Sets or queries the stop frequency (right edge) of the audio spectrum graph.
DISPlay:AUDio:SPECtrum:FREQuency:AUTO (on page 677)	Rescales the horizontal axis automatically to fit the waveform to the screen in the audio spectrum view.
DISPlay:AUDio:SPECtrum:MARKer:SHOW:STATe (on page 679)	Determines whether to show or hide the readouts for the selected marker in the audio spectrum view.
DISPlay:AUDio:SPECtrum:RESet:SCALe (on page 679)	Resets the horizontal and vertical scales in the audio spectrum view.
DISPlay:AUDio:SPECtrum:SCALe:LOG:STATe (on page 679)	Determines whether or not to set the horizontal axis logarithmic in the audio spectrum view.
DISPlay:AUDio:SPECtrum:SHOW:NHARmonic:THReshold (on page 680)	Sets or queries the appearance of the non-harmonic threshold in the audio spectrum view.
DISPlay:AUDio:SPECtrum:TABLe:SHOW:STATe (on page 680)	Determines whether to show or hide the results table in the audio spectrum view.
DISPlay:AUDio:SPECtrum:WINDow:TRACe:GRATicule:GRID:STATe (on page 681)	Determines whether to show or hide the graticule grid on the screen.
DISPlay:AUDio:SPECtrum:Y:[SCALe] (on page 681)	Sets or queries the vertical scale in the audio spectrum view.

Command	Description
DISPlay:AUDio:SPECtrum:Y:[SCALe]:AUTO (on page 682)	Rescales the vertical axis automatically in the audio spectrum view.
DISPlay:AUDio:SPECtrum:Y:[SCALe]:OFFSet (on page 682)	Sets or queries the vertical position (offset) in the audio spectrum view.
DISPlay:AVTime subgroup	**Amplitude versus Time measurement**
DISPlay:AVTime[:MEASview<y>]:LEGend:STATe (on page 682)	Shows or hides the trace legend in the amplitude versus time view.
DISPlay:AVTime[:MEASview<y>]:MARKer (on page 686):SHOW: STATe (on page 683)	Determines whether to show the readout for the selected marker.
DISPlay:AVTime[:MEASview<y>]:SELect:FIRSt (on page 683)	Queries the trace whose measurement results are being displayed in the readout on the top and bottom of the view.
DISPlay:AVTime[:MEASview<y>]:SELect:SECond (on page 684)	Sets or queries the trace whose measurement results are being displayed in the readout on the top and bottom of the view.
DISPlay:AVTime[:MEASview<y>]:SELect:THIRd (on page 684)	Queries the trace whose measurement results are being displayed in the readout on the top and bottom of the view.
DISPlay:AVTime[:MEASview<y>]:SELect:FOURth (on page 685)	Queries the trace whose measurement results are being displayed in the readout on the top and bottom of the view.
DISPlay:AVTime[:MEASview<y>]:SELect: TIMestamp (on page 686)	Sets or queries the display of the time stamp in the ogram plot.
DISPlay:AVTime[:MEASview<y>]:READout[:STATe] (on page 686)	Determines whether to show or hide the measurement parameter readouts in the Amplitude versus Time view.
DISPlay:AVTime[:MEASview<y>]:RESet (on page 687)	Resets the horizontal and vertical scale to the default values.
DISPlay:AVTime[:MEASview<y>]:TRIGger:LEVel: STATe (on page 687)	Determines whether to show the power trigger level line on screen.
DISPlay:AVTime[:MEASview<y>]:WINDow:TRACe: GRATicule:GRID:STATe (on page 688)	Determines whether to show the graticule grid on screen.
DISPlay:AVTime[:MEASview<y>]:X:RSCale (on page 688)	Rescales the horizontal axis automatically.
DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:AUTO (on page 689)	Sets the horizontal scale automatically.
DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:AUTO: STATe (on page 689)	Determines whether to set the horizontal scale automatically or manually.
DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:FULL (on page 690)	Sets or queries the horizontal scale.

Command	Description
DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:MAXimum? (on page 690)	Queries the upper limit of the horizontal scale setting range.
DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:MINimum? (on page 691)	Queries the lower limit of the horizontal scale setting range.
DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:OFFSet (on page 691)	Sets or queries the minimum horizontal value (left edge).
DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MAXimum? (on page 692)	Queries the upper limit of the horizontal offset setting range.
DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MINimum? (on page 693)	Queries the lower limit of the horizontal offset setting range.
DISPlay:AVTime[:MEASview<y>]:Y:RSCale (on page 693)	Rescales the vertical axis automatically.
DISPlay:AVTime[:MEASview<y>]:Y[:SCALe]:FULL (on page 694)	Sets or queries the vertical scale.
DISPlay:AVTime[:MEASview<y>]:Y[:SCALe]: OFFSet (on page 694)	Sets or queries the vertical offset.
DISPlay:CONSte subgroup (Option 21 only)	**Constellation measurement**
DISPlay:CONSte[:MEASview<x>]:MPHase (on page 707)	Sets or queries the multiplication constant of the phase multiplication constellation display for a CPM signal. This command is valid when `[SENSe]:DDEMod:MODulation:TYPE` is set to CPM.
DISPlay:CONSte[:MEASview<x>]:WINDow:TRACe:GRATicule:GRID:STATe (on page 708)	Determines whether to show or hide the graticule grid on the screen.
DISPlay:DDEMod subgroup (Option 21 only)	**General purpose digital modulation measurements**
DISPlay:DDEMod[:MEASview<y>]:DELete (on page 709)	Deletes the measurement view in the general purpose digital modulation measurements.
DISPlay:DDEMod[:MEASview<y>][:SOURce<x>]: NEW (on page 710)	Displays a new measurement view.
DISPlay:DDEMod[:MEASview<y>]:SELect (on page 710)	Selects a measurement view in the general purpose digital modulation measurements on the screen.
DISPlay:DDEMod[:SOURce<x>]:RADix (on page 711)	Sets or queries the base of symbols of the measurement source.
DISPlay:DDEMod[:SOURce<x>]:X[:SCALe] (on page 711)	Sets or queries the horizontal scale.
DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:AUTO (on page 712)	Sets the horizontal scale automatically.
DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:AUTO:STATe (on page 712)	Determines whether to set the horizontal scale automatically or manually.

Command	Description
DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:MAXimum? (on page 713)	Queries the upper limit of the horizontal scale setting range.
DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]: MINimum? (on page 713)	Queries the lower limit of the horizontal scale setting range.
DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:OFFSet (on page 714)	Sets or queries the minimum horizontal value (left edge).
DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:OFFSet:MAXimum? (on page 715)	Queries the upper limit of the horizontal offset setting range.
DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:OFFSet:MINimum? (on page 715)	Queries the lower limit of the horizontal offset setting range.
DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:RESet (on page 715)	Presets the horizontal scale to the default value.
DISPlay:DIAGram subgroup (Option 21 only)	**Eye/Trellis diagram**
DISPlay:DIAGram:X[:SCALe] (on page 716)	Sets or queries the horizontal range.
DISPlay:DIAGram:X[:SCALe]:RESet (on page 716)	Presets the horizontal scale to the default value.
DISPlay:DIQVtime subgroup (Option 21 only)	**Demodulated I&Q versus Time measurement**
DISPlay:DIQVtime[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe (on page 717)	Sets or queries whether to show the graticule grid on the screen.
DISPlay:DIQVtime[:MEASview<y>]:Y[:SCALe] (on page 717)	Sets or queries the vertical scale.
DISPlay:DIQVtime[:MEASview<y>]:Y[:SCALe]:AUTO (on page 718)	Sets the vertical scale automatically.
DISPlay:DIQVtime[:MEASview<y>]:Y[:SCALe]:OFFSet (on page 718)	Sets or queries the vertical offset (center point of the vertical axis).
DISPlay:DPX subgroup	**DPXogram measurement**
DISPlay:DPX:DGRam:TIME[:SCALe]:OFFSet:DIVision (on page 719)	Sets or queries the DPXogram vertical time offset in divisions.
DISPlay:DPX:DGRam:TIME[:SCALe]:PDIVision (on page 719)	Sets or queries the DPXogram vertical time scale per division.
DISPlay:DPX:DGRam:TIME[:SCALe]:RESet (on page 720)	Presets the time scale to the default value for the DPXogram measurement.
DISPlay:DPX:DGRam:TSTamp:STATe (on page 720)	Determines whether to display the time stamp readout in the DPXogram display.
DISPlay:DPX:DGRam:Y[:SCALe]:AUTO (on page 721)	Rescales the height axis automatically to fit the waveform to the screen in the DPXogram display.

Command	Description
DISPlay:DPX:DGRam:Y[:SCALe]:RESet (on page 721)	Resets the height scale of the DPXogram display.
DISPlay:DPX:LEGend:STATe (on page 722)	Determines whether to show or hide the trace legend on the display.
DISPlay:DPX:PHASe:Y[:SCALe]:AXIS (on page 722)	Sets or queries the vertical axis representation.
DISPlay:DPX:PHASe:Y[:SCALe]:AXIS:OFFSet (on page 723)	Sets or queries the vertical offset in the DPX Phase view.
DISPlay:DPX:Y[:SCALe]:OFFSet (on page 724)	Sets or queries the vertical offset.
DISPlay:DPX:Y[:SCALe]:PDIVision (on page 724)	Sets or queries the vertical scale (per division).
DISPlay:DPX:WINDow:TRACe:GRATicule:GRID:STATe (on page 723)	Shows or hides the graticule grid on the screen
DISPlay:EDIagram subgroup (Option 21 only)	**Eye diagram**
DISPlay:EDIagram[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe (on page 725)	Determines whether to show or hide the graticule grid on the screen in the eye diagram.
DISPlay:EDIagram[:MEASview<y>]:Y[:SCALe] (on page 725)	Sets or queries the vertical range of the eye diagram.
DISPlay:EDIagram[:MEASview<y>]:Y[:SCALe]: AUTO (on page 726)	Sets the vertical scale automatically to fit the waveform to the screen in the eye diagram.
DISPlay:EDIagram[:MEASview<y>]:Y[:SCALe]: OFFSet (on page 726)	Sets or queries the vertical offset (center point of the vertical axis) of the eye diagram.
DISPlay:EVM subgroup (Option 21 only)	**EVM versus Time measurement**
DISPlay:EVM[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe (on page 735)	Sets or queries the graticule grid view state for the EVM view.
DISPlay:EVM[:MEASview<y>]:Y[:SCALe] (on page 736)	Sets or queries the vertical range of the EVM versus Time graph.
DISPlay:EVM[:MEASview<y>]:Y[:SCALe]:AUTO (on page 737)	Sets the vertical scale automatically to fit the waveform to the screen in the EVM versus Time display.
DISPlay:EVM[:MEASview<y>]:Y[:SCALe]:OFFSet (on page 737)	Sets or queries the minimum vertical value (bottom edge) of the EVM versus Time graph.
DISPlay:FDVTime subgroup (Option 21 only)	**Frequency deviation versus Time measurement**
DISPlay:FDVTime[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe (on page 738)	Sets or queries the graticule grid view state for the FDV view.
DISPlay:FDVTime[:MEASview<y>]:Y[:SCALe] (on page 738)	Sets or queries the vertical range of the Frequency deviation versus Time graph.

Command	Description
DISPlay:FDVTime[:MEASview<y>]:Y[:SCALe]:AUTO (on page 739)	Sets the vertical scale automatically to fit the waveform to the screen in the Frequency deviation versus Time display.
DISPlay:FDVTime[:MEASview<y>]:Y[:SCALe]: OFFSet (on page 740)	Sets or queries the minimum vertical value (bottom edge) of the Frequency deviation versus Time graph.
DISPlay:{FSETtling|PSETtling} subgroup	**Frequency and Phase Settling measurements**
DISPlay:{FSETtling|PSETtling}:MARKer:SHOW: STATe (on page 740)	Determines whether to show or hide the readout for the selected marker in the Frequency or Phase Settling views.
DISPlay:{FSETtling|PSETtling}:TIME:DECimal (on page 741)	Sets or queries the number of values to the right of the decimal point.
DISPlay:{FSETtling|PSETtling}:WINDow:TRACe:GRATicule:GRID:STATe (on page 741)	Determines whether to show or hide the graticule grid on the screen.
DISPlay:{FSETtling|PSETtling}:X[:SCALe] (on page 742)	Sets or queries the horizontal scale (full-scale time) of the Settling Time graph.
DISPlay:{FSETtling|PSETtling}:X[:SCALe]:AUTO (on page 742)	Sets the horizontal scale automatically to fit the waveform to the screen in the Frequency and Phase Settling views.
DISPlay:{FSETtling|PSETtling}:X[:SCALe]:AUTO:STATe (on page 743)	Determines whether to set the horizontal scale automatically or manually.
DISPlay:{FSETtling|PSETtling}:X[:SCALe]: MAXimum? (on page 743)	Queries the upper limit of the horizontal scale setting range.
DISPlay:{FSETtling|PSETtling}:X[:SCALe]:MINimum? (on page 744)	Queries the lower limit of the horizontal scale setting range.
DISPlay:{FSETtling|PSETtling}:X[:SCALe]:OFFSet:MINimum? (on page 745)	Queries the lower limit of the horizontal offset setting range.
DISPlay:{FSETtling|PSETtling}:X[:SCALe]:OFFSet (on page 744)	Sets or queries the minimum horizontal value (left edge) of the Frequency versus Time graph.
DISPlay:{FSETtling|PSETtling}:X[:SCALe]:OFFSet:MAXimum? (on page 745)	Queries the upper limit of the horizontal offset setting range.
DISPlay:{FSETtling|PSETtling}:X[:SCALe]:OFFSet:MINimum? (on page 745)	Queries the lower limit of the horizontal offset setting range.
DISPlay:{FSETtling|PSETtling}:Y[:SCALe] (on page 746)	Sets or queries the vertical range of the Frequency versus Time graph.
DISPlay:{FSETtling|PSETtling}:Y[:SCALe]:AUTO (on page 746)	Sets the vertical scale automatically to fit the waveform to the screen in the Frequency and Phase Settling views.
DISPlay:{FSETtling|PSETtling}:Y[:SCALe]:OFFSet (on page 747)	Sets or queries the vertical offset (the value at the center of the vertical axis) in the Frequency and Phase Settling graphs.

Command	Description
DISPlay:{FSETtling|PSETtling}:Y[:SCALe]:PDIVision (on page 747)	Sets or queries the vertical scale (per division) of the Frequency and Phase Settling graphs.
DISPlay:FVTime subgroup	**Frequency versus Time measurement**
DISPlay:FVTime[:MEASview<y>]:READout[:STATe] (on page 748)	Determines whether to show or hide the measurement parameter readouts in the Frequency versus Time view.
DISPlay:FVTime[:MEASview<y>]:SELected:TIMestamp (on page 748)	Sets or queries the display of the time stamp in the ogram plot.
DISPlay:FVTime[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe (on page 749)	Sets or queries whether to show the graticule grid on the Frequency versus Time view.
DISPlay:FVTime[:MEASview<y>]:X[:SCALe] (on page 749)	Sets or queries the horizontal scale.
DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:AUTO (on page 750)	Sets the horizontal scale automatically.
DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:AUTO:STATe (on page 751)	Determines whether to set the horizontal scale automatically or manually.
DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:MAXimum? (on page 751)	Queries the upper limit of the horizontal scale setting range.
DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:MINimum? (on page 752)	Queries the lower limit of the horizontal scale setting range.
DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:OFFSet (on page 752)	Sets or queries the minimum horizontal value (left edge).
DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MAXimum? (on page 753)	Queries the upper limit of the horizontal offset setting range.
DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MINimum? (on page 754)	Queries the lower limit of the horizontal offset setting range.
DISPlay:FVTime[:MEASview<y>]:Y[:SCALe] (on page 754)	Sets or queries the vertical scale.
DISPlay:FVTime[:MEASview<y>]:Y[:SCALe]:AUTO (on page 755)	Sets the vertical scale automatically.
DISPlay:FVTime[:MEASview<y>]:Y[:SCALe]:OFFSet (on page 755)	Sets or queries the vertical offset.
DISPlay:GENeral subgroup	**General signal viewing**
DISPlay:GENeral:MEASview<y>:DELete (on page 756)	Deletes the measurement view.
DISPlay:GENeral:MEASview<y>[:SOURce<x>]: NEW (on page 757)	Displays a new measurement view.

Command	Description
DISPlay:GENeral:MEASview<y>:SELect (on page 757)	Sets or queries the measurement view.
DISPlay:GPRF subgroup	**General purpose RF measurements**
DISPlay:GPRF:MEASview:DELete (on page 758)	Deletes the measurement view.
DISPlay:GPRF:MEASview[:SOURce<x>]:NEW (on page 759)	Displays a new measurement view.
DISPlay:GPRF:MEASview:SELect (on page 760)	Sets or queries the measurement view.
DISPlay:IQVTime subgroup	**RF I&Q versus Time measurement**
DISPlay:IQVTime[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe (on page 761)	Sets or queries whether to show the graticule grid on the screen.
DISPlay:IQVTime[:MEASview<y>]:X[:SCALe] (on page 761)	Sets or queries the horizontal scale.
DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:AUTO (on page 762)	Sets the horizontal scale automatically.
DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:AUTO:STATe (on page 763)	Determines whether to set the horizontal scale automatically or manually.
DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:MAXimum? (on page 763)	Queries the upper limit of the horizontal scale setting range.
DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:MINimum? (on page 764)	Queries the lower limit of the horizontal scale setting range.
DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]: OFFSet (on page 764)	Sets or queries the minimum horizontal value (left edge).
DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MAXimum? (on page 765)	Queries the upper limit of the horizontal offset setting range.
DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MINimum? (on page 766)	Queries the lower limit of the horizontal offset setting range.
DISPlay:IQVTime[:MEASview<y>]:Y[:SCALe] (on page 766)	Sets or queries the vertical scale.
DISPlay:IQVTime[:MEASview<y>]:Y[:SCALe]:AUTO (on page 767)	Sets the vertical scale automatically.
DISPlay:IQVTime[:MEASview<y>]:Y[:SCALe]: OFFSet (on page 767)	Sets or queries the vertical offset.
DISPlay:IQVTime[:MEASview<y>]:Y[:SCALe]: RESCale (on page 768)	Rescales the vertical scale.
DISPlay:MCPower subgroup	**MCPR measurement**

Command	Description
DISPlay:MCPower:MARKer:SHOW:STATe (on page 785)	Determines whether to show the readout for the selected marker.
DISPlay:MCPower:PLEVel:SHOW:STATe (on page 785)	Determines whether to show the power levels.
DISPlay:MCPower:RESet:SCALe (on page 785)	Resets the horizontal and vertical scale to the default values.
DISPlay:MCPower:WINDow:TRACe:GRATicule:GRID:STATe (on page 786)	Determines whether to show the graticule grid on screen.
DISPlay:MCPower:X[:SCALe] (on page 786)	Sets or queries the horizontal range.
DISPlay:MCPower:X[:SCALe]:AUTO (on page 787)	Rescales the horizontal axis automatically.
DISPlay:MCPower:X[:SCALe]:OFFSet (on page 787)	Sets or queries the minimum horizontal value (left edge).
DISPlay:MCPower:Y[:SCALe] (on page 788)	Sets or queries the vertical range.
DISPlay:MCPower:Y[:SCALe]:AUTO (on page 788)	Rescales the vertical axis automatically.
DISPlay:MCPower:Y[:SCALe]:OFFSet (on page 789)	Sets or queries the vertical offset.
DISPlay:MERRor subgroup (Option 21 only)	**Magnitude error versus Time measurement**
DISPlay:MERRor[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe (on page 789)	Sets or queries the graticule grid view state for the Magnitude error versus Time view.
DISPlay:MERRor[:MEASview<y>]:Y[:SCALe] (on page 790)	Sets or queries the vertical range of the Magnitude error versus Time graph.
DISPlay:MERRor[:MEASview<y>]:Y[:SCALe]:AUTO (on page 790)	Sets the vertical scale automatically to fit the waveform to the screen in the Magnitude error versus Time display.
DISPlay:MERRor[:MEASview<y>]:Y[:SCALe]: OFFSet (on page 791)	Sets or queries the minimum vertical value (bottom edge) of the Magnitude error versus Time graph.
DISPlay:OBWidth subgroup	**Occupied Bandwidth measurement**
DISPlay:OBWidth:MARKer:SHOW:STATe (on page 821)	Determines whether to show the readout for the selected marker.
DISPlay:OBWidth:RESet:SCALe (on page 821)	Resets the horizontal and vertical scale to the default values.
DISPlay:OBWidth:SELected:BANDwidth (on page 822)	Sets or queries the bandwidth (OBW or x dB BW) to measure.
DISPlay:OBWidth:WINDow:TRACe:GRATicule:GRID:STATe (on page 822)	Determines whether to show the graticule grid on screen.
DISPlay:OBWidth:X[:SCALe] (on page 822)	Sets or queries the horizontal range.
DISPlay:OBWidth:X[:SCALe]:AUTO (on page 823)	Rescales the horizontal axis automatically.

Command	Description
DISPlay:OBWidth:X[:SCALe]:OFFSet (on page 823)	Sets or queries the minimum horizontal value (left edge).
DISPlay:OBWidth:Y[:SCALe] (on page 824)	Sets or queries the vertical range
DISPlay:OBWidth:Y[:SCALe]:AUTO (on page 824)	Rescales the vertical axis automatically.
DISPlay:OBWidth:Y[:SCALe]:OFFSet (on page 825)	Sets or queries the vertical offset.
DISPlay:OFDM subgroup	**OFDM measurements**
DISPlay:OFDM:CONSte[:SCALe] (on page 825)	Sets or queries the horizontal offset.
DISPlay:OFDM:CONSte:X:OFFSet (on page 826)	Sets or queries the vertical offset.
DISPlay:OFDM:CONSte:Y:OFFSet (on page 826)	Sets or queries the scale.
DISPlay:OFDM:CRESponse:FREQuency:AUTO (on page 826)	Rescales the horizontal axis automatically.
DISPlay:OFDM:CRESponse:FREQuency:OFFSet (on page 827)	Sets or queries the frequency offset.
DISPlay:OFDM:CRESponse:MAGNitude:AUTO (on page 828)	Sets or queries the horizontal range.
DISPlay:OFDM:CRESponse:MAGNitude:OFFSet (on page 828)	Rescales the magnitude automatically.
DISPlay:OFDM:CRESponse:MAGNitude:PDIVision (on page 828)	Sets or queries the magnitude offset.
DISPlay:OFDM:CRESponse:PHASe:AUTO (on page 829)	Sets or queries the scale (per division) of the magnitude.
DISPlay:OFDM:CRESponse:PHASe:OFFSet (on page 829)	Rescales the vertical axis automatically.
DISPlay:OFDM:CRESponse:PHASe[:SCALe] (on page 830)	Sets or queries the phase offset.
DISPlay:OFDM:CRESponse:WINDow:SELect:PLOT (on page 830)	Sets the vertical scale automatically.
DISPlay:OFDM:CRESponse:WINDow:TRACe:GRATicule:GRID:STATe (on page 830)	Sets or queries the plot
DISPlay:OFDM:EVM:FREQuency[:SCALe] (on page 832)	Determines whether to show or hide the graticule.
DISPlay:OFDM:EVM:FREQuency:AUTO (on page 831)	Sets or queries the horizontal range.

Command	Description
DISPlay:OFDM:EVM:FREQuency:OFFSet (on page 831)	Rescales the frequency automatically to fit the waveform to the screen.
DISPlay:OFDM:EVM:MARKer:SHOW:STATe (on page 832)	Sets or queries the frequency offset.
DISPlay:OFDM:EVM:TIME[:SCALe] (on page 833)	Determines whether to show or hide the readout.
DISPlay:OFDM:EVM:TIME:AUTO (on page 832)	Rescales the time automatically to fit the waveform to the screen..
DISPlay:OFDM:EVM:TIME:OFFSet (on page 833)	Sets or queries the time offset.
DISPlay:OFDM:EVM:WINDow:SELect:PLOT (on page 834)	Sets or queries the plot.
DISPlay:OFDM:EVM:WINDow:TRACe:GRATicule:GRID:STATe (on page 834)	Determines whether to show or hide the graticule.
DISPlay:OFDM:EVM:Y[:SCALe] (on page 835)	Sets or queries the vertical range.
DISPlay:OFDM:EVM:Y:AUTO (on page 834)	Rescales the vertical scale automatically.
DISPlay:OFDM:EVM:Y:OFFSet (on page 835)	Sets or queries the vertical offset.
DISPlay:OFDM:FLATness:AUTO (on page 836)	Automatically rescales the horizontal and vertical axes for the best display.
DISPlay:OFDM:FLATness:X:AUTO (on page 836)	Rescales the horizontal axis automatically in the OFDM Spectral Flatness display.
DISPlay:OFDM:FLATness:X:OFFSet (on page 836)	Sets or queries the value of the offset of the horizontal axis.
DISPlay:OFDM:FLATness:X[:SCALe] (on page 837)	Sets or queries the value of the horizontal scale.
DISPlay:OFDM:FLATness:Y:AUTO (on page 837)	Rescales the vertical axis scale and position values automatically.
DISPlay:OFDM:FLATness:Y:OFFSet (on page 838)	Sets or queries the value of the offset from the center (vertical position).
DISPlay:OFDM:FLATness:Y[:SCALe] (on page 838)	Sets or queries the vertical scale value.
DISPlay:OFDM:MEASview:DELete (on page 839)	Deletes the specified OFDM measurement view.
DISPlay:OFDM:MEASview:NEW (on page 839)	Creates a new OFDM measurement view.
DISPlay:OFDM:MEASview:SELect (on page 840)	Sets or queries the OFDM measurement view.
DISPlay:OFDM:MERRor:FREQuency[:SCALe] (on page 841)	Sets or queries the horizontal range.
DISPlay:OFDM:MERRor:FREQuency[:SCALe]: AUTO (on page 841)	Rescales the frequency automatically to fit the waveform to the screen.

Command	Description
DISPlay:OFDM:MERRor:FREQuency[:SCALe]: OFFSet (on page 841)	Sets or queries the frequency offset.
DISPlay:OFDM:MERRor:MARKer:SHOW:STATe (on page 842)	Determines whether to show or hide the readout for the selected marker.
DISPlay:OFDM:MERRor:TIME[:SCALe] (on page 842)	Sets or queries the time scale.
DISPlay:OFDM:MERRor:TIME[:SCALe]:AUTO (on page 843)	Rescales the time automatically to fit the waveform to the screen.
DISPlay:OFDM:MERRor:TIME[:SCALe]:OFFSet (on page 843)	Sets or queries the time offset.
DISPlay:OFDM:MERRor:WINDow:SELect:PLOT (on page 843)	Sets or queries the plot.
DISPlay:OFDM:MERRor:WINDow:TRACe:GRATicule:GRID:STATe (on page 844)	Determines whether to show or hide the graticule.
DISPlay:OFDM:MERRor:Y[:SCALe] (on page 845)	Sets or queries the vertical range.
DISPlay:OFDM:MERRor:Y:AUTO (on page 844)	Rescales the vertical scale automatically to fit the waveform.
DISPlay:OFDM:MERRor:Y:OFFSet (on page 845)	Sets or queries the vertical offset.
DISPlay:OFDM:PERRor:FREQuency[:SCALe] (on page 845)	Sets or queries the horizontal range.
DISPlay:OFDM:PERRor:FREQuency[:SCALe]:AUTO (on page 846)	Rescales the frequency automatically to fit the waveform to the screen.
DISPlay:OFDM:PERRor:FREQuency[:SCALe]: OFFSet (on page 846)	Sets or queries the frequency offset.
DISPlay:OFDM:PERRor:MARKer:SHOW:STATe (on page 847)	Determines whether to show or hide the readout for the selected marker.
DISPlay:OFDM:PERRor:TIME[:SCALe] (on page 847)	Sets or queries the time scale.
DISPlay:OFDM:PERRor:TIME[:SCALe]:AUTO (on page 847)	Rescales the time automatically to fit the waveform to the screen.
DISPlay:OFDM:PERRor:TIME[:SCALe]:OFFSet (on page 848)	Sets or queries the time offset.
DISPlay:OFDM:PERRor:WINDow:SELect:PLOT (on page 848)	Sets or queries the plot.
DISPlay:OFDM:PERRor:WINDow:TRACe:GRATicule:GRID:STATe (on page 849)	Determines whether to show or hide the graticule.

Command	Description
DISPlay:OFDM:PERRor:Y[:SCALe] (on page 850)	Sets or queries the vertical range.
DISPlay:OFDM:PERRor:Y:AUTO (on page 849)	Rescales the vertical scale automatically to fit the waveform.
DISPlay:OFDM:PERRor:Y:OFFSet (on page 849)	Sets or queries the vertical offset.
DISPlay:OFDM:POWer:FREQuency[:AUTO] (on page 850)	Rescales the frequency automatically to fit the waveform to the screen.
DISPlay:OFDM:POWer:FREQuency[:OFFSet] (on page 851)	Sets or queries the frequency offset.
DISPlay:OFDM:POWer:FREQuency[:SCALe] (on page 851)	Sets or queries the horizontal range.
DISPlay:OFDM:POWer:MARKer:SHOW:STATe (on page 851)	Determines whether to show or hide the readout for the selected marker.
DISPlay:OFDM:POWer:TIME[:AUTO] (on page 852)	Rescales the time automatically to fit the waveform to the screen.
DISPlay:OFDM:POWer:TIME:OFFSet (on page 852)	Sets or queries the time offset.
DISPlay:OFDM:POWer:TIME[:SCALe] (on page 853)	Sets or queries the time scale.
DISPlay:OFDM:POWer:WINDow:SELect:PLOT (on page 853)	Sets or queries the plot.
DISPlay:OFDM:POWer:WINDow:TRACe:GRATicule:GRID:STATe (on page 853)	Determines whether to show or hide the graticule.
DISPlay:OFDM:POWer:Y:AUTO (on page 854)	Rescales the vertical scale automatically to fit the waveform.
DISPlay:OFDM:POWer:Y:OFFSet (on page 854)	Sets or queries the vertical offset.
DISPlay:OFDM:POWer:Y[:SCALe] (on page 855)	Sets or queries the vertical range.
DISPlay:PERRor subgroup (Option 21 only)	**Phase error versus Time measurement**
DISPlay:PERRor[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe (on page 869)	Sets or queries the graticule grid view state for the Phase error versus Time view.
DISPlay:PERRor[:MEASview<y>]:Y[:SCALe] (on page 870)	Sets or queries the vertical scale.
DISPlay:PERRor:[:MEASview<y>]Y[:SCALe]:AUTO (on page 871)	Sets the vertical scale automatically.
DISPlay:PERRor[:MEASview<y>]:Y[:SCALe]: OFFSet (on page 871)	Sets or queries the minimum vertical value (bottom edge).
DISPlay:PHVTime subgroup	**Phase versus Time measurement**

Command	Description
DISPlay:PHVTime[:MEASview<y>]:SELected:TIMestamp (on page 872)	Sets or queries the display of the time stamp in the ogram plot.
DISPlay:PHVTime[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe (on page 872)	Sets or queries whether to show the graticule grid on the screen.
DISPlay:PHVTime[:MEASview<y>]:X[:SCALe] (on page 873)	Sets or queries the horizontal scale.
DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:AUTO (on page 873)	Sets the horizontal scale automatically.
DISPlay:PHVTime[:MEASview<y>]:READout[:STATe] (on page 874)	Determines whether to show or hide the measurement parameter readouts in the Phase versus Time view.
DISPlay:PHVTime[:MEASview?]:Y[:SCALe]:AXIS:ENABle:REFerence (on page 874)	Sets or queries the state of **Y Reference** checkbox.
DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:AUTO:STATe (on page 875)	Determines whether to set the horizontal scale automatically or manually.
DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:MAXimum? (on page 875)	Queries the upper limit of the horizontal scale setting range.
DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:MINimum? (on page 876)	Queries the lower limit of the horizontal scale setting range.
DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]: OFFSet (on page 877)	Sets or queries the minimum horizontal value (left edge).
DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MAXimum? (on page 877)	Queries the upper limit of the horizontal offset setting range.
DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MINimum? (on page 878)	Queries the lower limit of the horizontal offset setting range.
DISPlay:PHVTime[:MEASview<y>]:Y[:SCALe] (on page 878)	Sets or queries the vertical scale.
DISPlay:PHVTime[:MEASview<y>]:Y[:SCALe]:AUTO (on page 879)	Sets the vertical scale automatically.
DISPlay:PHVTime[:MEASview<y>]:Y[:SCALe]:AXIS (on page 879)	Sets or queries the vertical axis representation.
DISPlay:PHVTime[:MEASview<y>]:Y[:SCALe]:AXIS:REFerence (on page 880)	Sets or queries the reference time for phase.
DISPlay:PHVTime[:MEASview<y>]:Y[:SCALe]: OFFSet (on page 880)	Sets or queries the vertical offset.
DISPlay:PHVTime[:MEASview<y>]:Y[:SCALe]:RESCale (on page 881)	Rescales the vertical scale.

Command	Description
DISPlay:PNOise subgroup (Option 11 only)	**Phase noise measurements**
DISPlay:PNOise:LEGend:STATe (on page 881)	Sets or queries showing the trace legend on the display.
DISPlay:PNOise:MARKer:SHOW:STATe (on page 882)	Determines whether to show the readout for the selected marker.
DISPlay:PNOise:RESet:SCALe (on page 882)	Resets the horizontal and vertical scale to the default values.
DISPlay:PNOise:WINDow:TRACe:GRATicule:GRID:STATe (on page 882)	Sets or queries whether to show the graticule grid on the screen.
DISPlay:PNOise:X[:SCALe]:AUTO (on page 883)	Rescales the horizontal axis automatically.
DISPlay:PNOise:X[:SCALe]:STARt (on page 883)	Sets or queries the start frequency of the graph.
DISPlay:PNOise:X[:SCALe]:STOP (on page 884)	Sets or queries the stop frequency of the graph.
DISPlay:PNOise:Y[:SCALe] (on page 884)	Sets or queries the stop frequency of the graph.
DISPlay:PNOise:Y[:SCALe]:AUTO (on page 884)	Rescales the vertical axis automatically.
DISPlay:PNOise:Y[:SCALe]:OFFSet (on page 885)	Sets or queries the vertical offset.
DISPlay:PNOise:Y[:SCALe]:PDIVision (on page 885)	Sets or queries the vertical scale (per division).
DISPlay:PULSe subgroup	**Pulsed RF measurements**
DISPlay:PULSe:MEASview<y>:DELete (on page 909)	Deletes the measurement view.
DISPlay:PULSe:MEASview<y>[:SOURce<x>]:NEW (on page 909)	Displays a new measurement view.
DISPlay:PULSe:MEASview<y>:SELect (on page 910)	Selects the specifies Measurement view.
DISPlay:PULSe:RESult[:MEASview<y>]:ATX (on page 920)	Sets or queries whether to show the average transmitted power result.
DISPlay:PULSe:RESult[:MEASview<y>]:AVERage (on page 921)	Sets or queries whether to show the average on power result.
DISPlay:PULSe:RESult[:MEASview<y>]:AVERage (on page 921)	Sets or queries whether to show the average on power result.
DISPlay:PULSe:RESult[:MEASview<y>]:DRODb (on page 921)	Sets or queries whether to show the droop in the results table.
DISPlay:PULSe:RESult[:MEASview<y>]:DROop (on page 922)	Sets or queries showing the droop measurement result in the pulse table
DISPlay:PULSe:RESult[:MEASview<y>]:DUTPct (on page 922)	Sets or queries whether to show the duty factor (%) result.

Command	Description
DISPlay:PULSe:RESult[:MEASview<y>]:DUTRatio (on page 923)	Sets or queries whether to show the duty factor (ratio) result.
DISPlay:PULSe:RESult[:MEASview<y>]:FALL (on page 924)	Sets or queries whether to show the fall time in the results table.
DISPlay:PULSe:RESult[:MEASview<y>]:FDELta (on page 924)	Sets or queries whether to show the delta frequency result.
DISPlay:PULSe:RESult[:MEASview<y>]:FRDeviation (on page 925)	Sets or queries whether to show the frequency deviation result.
DISPlay:PULSe:RESult[:MEASview<y>]:IRAMplitude (on page 925)	Sets or queries showing the Impulse Response Amplitude measurement result in the pulse table.
DISPlay:PULSe:RESult[:MEASview<y>]:IRTime (on page 926)	Sets or queries showing the Impulse Response Time measurement result in the pulse table.
DISPlay:PULSe:RESult[:MEASview<y>]:MFReqerror (on page 926)	Sets or queries whether to show the maximum frequency error result.
DISPlay:PULSe:RESult[:MEASview<y>]:MPHerror (on page 927)	Sets or queries whether to show the maximum phase error result.
DISPlay:PULSe:RESult[:MEASview<y>]:OVEDb (on page 927)	Sets or queries whether to show the Overshoot measurement result in dB in the pulse table.
DISPlay:PULSe:RESult[:MEASview<y>]:OVERshoot (on page 928)	Sets or queries whether to show the Overshoot measurement result in the pulse table.
DISPlay:PULSe:RESult[:MEASview<y>]:PHDeviation (on page 928)	Sets or queries whether to show the phase deviation result.
DISPlay:PULSe:RESult[:MEASview<y>]: PPFRequency (on page 930)	Sets or queries whether to show the pulse-pulse frequency result.
DISPlay:PULSe:RESult[:MEASview<y>]:PPOWer (on page 930)	Sets or queries whether to show the peak power in the results table.
DISPlay:PULSe:RESult[:MEASview<y>]:PPPHase (on page 931)	Sets or queries whether to show the pulse-pulse carrier phase result.
DISPlay:PULSe:RESult[:MEASview<y>]:RINTerval (on page 931)	Sets or queries whether to show the repetition interval result.
DISPlay:PULSe:RESult[:MEASview<y>]:RIPDb (on page 932)	Sets or queries showing the ripple measurement result in dB in the pulse table.
DISPlay:PULSe:RESult[:MEASview<y>]:RIPPle (on page 932)	Sets or queries showing the ripple measurement result in dB in the pulse table.
DISPlay:PULSe:RESult[:MEASview<y>]:RISE (on page 933)	Sets or queries whether to show the rise time in the results table.

Command	Description
DISPlay:PULSe:RESult[:MEASview<y>]: RMSFreqerror (on page 933)	Sets or queries whether to show the RMS frequency error result.
DISPlay:PULSe:RESult[:MEASview<y>]:RMSPherror (on page 934)	Sets or queries whether to show the RMS phase error result.
DISPlay:PULSe:RESult[:MEASview<y>]:RRATe (on page 934)	Sets or queries whether to show the repetition rate result.
DISPlay:PULSe:RESult[:MEASview<y>]:TIME (on page 935)	Sets or queries whether to show the time in the results table.
DISPlay:PULSe:RESult[:MEASview<y>]:WIDTh (on page 935)	Sets or queries whether to show the pulse width in the results table.
DISPlay:PULSe:SELect[:MEASview<y>]:NUMBer (on page 936)	Sets or queries the pulse to measure.
DISPlay:PULSe:SELect[:MEASview<y>]:RESult (on page 936)	Sets or queries which result is shown in the trace and statistics views.
DISPlay:PULSe:STATistics[:MEASview<y>]: MARKer<x>:SHOW:STATe (on page 939)	Sets or queries whether to show the marker readout in the graph.
DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT (on page 940)	Sets or queries how to show the statistics graph.
DISPlay:PULSe:STATistics[:MEASview<y>]: WINDow:TRACe:GRATicule:GRID:STATe (on page 940)	Sets or queries whether to show the graticule grid in the statistics view.
DISPlay:PULSe:STATistics[:MEASview<y>]:X: RSCale (on page 941)	Rescales the horizontal axis of the statistics graph.
DISPlay:PULSe:STATistics[:MEASview<y>]: X[:SCALe]:NUMBer (on page 941)	Sets or queries the horizontal scale (the number of pulses per division).
DISPlay:PULSe:STATistics[:MEASview<y>]: X[:SCALe]:OFFSet (on page 942)	Sets or queries the minimum horizontal value in the statistics view.
DISPlay:PULSe:STATistics[:MEASview<y>]:Y: RSCale (on page 943)	Rescales the vertical axis of the statistics graph.
DISPlay:PULSe:STATistics[:MEASview<y>]: Y[:SCALe]:FULL (on page 943)	Sets or queries the vertical full-scale in the statistics view.
DISPlay:PULSe:STATistics[:MEASview<y>]: Y[:SCALe]:OFFSet (on page 944)	Sets or queries the vertical offset in the statistics view.
DISPlay:PULSe:STATistics[:MEASview<y>]: Y[:SCALe]:STOP? (on page 945)	Queries the minimum vertical value in the statistics view.

Command	Description
DISPlay:PULSe:TRACe[:MEASview<y>]:MARKer<x>:SHOW:STATe (on page 946)	Sets or queries whether to show the marker readout in the trace view.
DISPlay:PULSe:TRACe[:MEASview<y>]:POINt: SHOW (on page 946)	Sets or queries whether to show the measurement points and lines.
DISPlay:PULSe:TRACe[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe (on page 947)	Sets or queries whether to show the graticule grid in the trace view.
DISPlay:PULSe:TRACe[:MEASview<y>]:X:RSCale (on page 947)	Rescales the horizontal axis of the pulse trace view.
DISPlay:PULSe:TRACe[:MEASview<y>]:X[:SCALe] (on page 948)	Sets or queries the horizontal full scale in the pulse trace view.
DISPlay:PULSe:TRACe[:MEASview<y>]:X[:SCALe]:FULL (on page 948)	Sets or queries the full-scale reference for the horizontal rescale.
DISPlay:PULSe:TRACe[:MEASview<y>]:X[:SCALe]:OFFSet (on page 949)	Sets or queries the minimum horizontal value in the pulse trace view.
DISPlay:PULSe:TRACe[:MEASview<y>]:X[:SCALe]:PDIVision (on page 949)	Sets or queries the horizontal full scale in the pulse trace view.
DISPlay:PULSe:TRACe[:MEASview<y>]:Y:RSCale (on page 950)	Rescales the vertical axis of the pulse trace view.
DISPlay:PULSe:TRACe[:MEASview<y>]:Y[:SCALe]:FULL (on page 950)	Sets or queries the vertical full scale in the pulse trace view.
DISPlay:PULSe:TRACe[:MEASview<y>]:Y[:SCALe]:OFFSet (on page 951)	Sets or queries the vertical offset in the pulse trace view.
DISPlay:PULSe:TRACe[:MEASview<y>]:Y[:SCALe]:STOP? (on page 951)	Queries the minimum vertical value in the pulse trace view.
DISPlay:SEM subgroup	**Spectral Emissions Mask measurements**
DISPlay:SEM:MARKer:SHOW:STATe (on page 952)	Determines whether to show or hide the readout for the selected marker.
DISPlay:SEM:RESet:SCALe (on page 952)	Resets the scale.
DISPlay:SEM:SHOW:LIMit (on page 953)	Sets or queries the appearance of the limits.
DISPlay:SEM:WINDow:TRACe:GRATicule:GRID:STATe (on page 953)	Determines whether to show or hide the graticule.
DISPlay:SEM:X[:SCALe]:AUTO (on page 954)	Rescales the horizontal scale automatically to fit the waveform.
DISPlay:SEM:X[:SCALe]:START (on page 954)	Sets or queries the start frequency.
DISPlay:SEM:X[:SCALe]:STOP (on page 954)	Sets or queries the stop frequency.

Command	Description
DISPlay:SEM:Y[:SCALe] (on page 955)	Sets or queries the vertical range.
DISPlay:SEM:Y[:SCALe]:AUTO (on page 955)	Rescales the vertical scale automatically to fit the waveform.
DISPlay:SEM:Y[:SCALe]:OFFSet (on page 956)	Sets or queries the vertical offset.
DISPlay:SGRam subgroup	**Spectrogram measurement**
DISPlay:SGRam[:MEASview<y>]:FREQuency:AUTO (on page 956)	Rescales the horizontal (frequency) axis automatically.
DISPlay:SGRam[:MEASview<y>]:FREQuency: OFFSet (on page 956)	Sets or queries the horizontal (frequency) offset.
DISPlay:SGRam[:MEASview<y>]:FREQuency: SCAle (on page 957)	Sets or queries the horizontal (frequency) range.
DISPlay:SGRAM[:MEASview<y>]:MARKer:SHOW: STATe (on page 958)	Sets or queries the marker readout.
DISPlay:SGRAM[:MEASview<y>]:SELected: TIMestamp (on page 958)	Sets or queries the time stamp readout.
DISPlay:SGRam[:MEASview<y>]:TIME:AUTO (on page 959)	Rescales the vertical axis automatically.
DISPlay:SGRam[:MEASview<y>]:TIME:OFFSet (on page 959)	Sets or queries the vertical axis (time) offset (bottom line number).
DISPlay:SGRam[:MEASview<y>]:TIME:OFFSet: DIVisions (on page 960)	Sets or queries the Vertical offset or Position in divisions.
DISPlay:SGRam[:MEASview<y>]:TIME:OVERlap: PERCent (on page 961)	Sets or queries the Overlap in percent.
DISPlay:SGRam[:MEASview<y>]:TIME[:SCAle] (on page 961)	Sets or queries the vertical scale (the amount of time in each line).
DISPlay:SGRam[:MEASview<y>]:TIME:SCALe:PER: DIVision (on page 962)	Sets or queries the vertical Time/division value.
DISPlay:SGRam[:MEASview<y>]:TIME:SPECtrums: PERLine? (on page 963)	Queries the Spectrums/Line value.
DISPlay:SPECtrum subgroup	**Spectrum measurements**
DISPlay:SPECtrum[:MEASview<y>]:FREQuency: AUTO (on page 963)	Rescales the horizontal axis automatically to fit the waveform to the screen in the spectrum view.
DISPlay:SPECtrum[:MEASview<y>]:FREQuency: OFFSet (on page 963)	Sets or queries the frequency offset (the value at the center of the horizontal axis) in the spectrum graph.

Command	Description
DISPlay:SPECtrum[:MEASview<y>]:FREQuency [:SCAle] (on page 964)	Sets or queries the horizontal range of the spectrum graph.
DISPlay:SPECtrum[:MEASview<y>]:MARKer: NOISe:MODE (on page 965)	Determines whether to enable or disable the marker noise mode in the spectrum view.
DISPlay:SPECtrum[:MEASview<y>]:READout[:STATe] (on page 965)	Determines whether to enable or disable the marker noise mode in the spectrum view.
DISPlay:SPECtrum[:MEASview<y>][:SCALe]: LOG:STATe (on page 966)	Determines whether or not to set the horizontal axis logarithmic in the Spectrum view.
DISPlay:SPECtrum[:MEASview<y>]:WINDow:TRACe: GRATicule:GRID:STATe (on page 966)	Determines whether to show or hide the graticule grid on the screen.
DISPlay:SPECtrum[:MEASview<y>]:WINDow :TRACeLEGend:STATe (on page 967)	Determines whether to show or hide the trace legend in the Spectrum view.
DISPlay:SPECtrum[:MEASview<y>]:Y[:SCAle]:AUTO (on page 968)	Rescales the vertical axis automatically to fit the waveform to the screen in the Spectrum view.
DISPlay:SPECtrum[:MEASview<y>]:X:LABel (on page 967)	Sets or queries the labels for the horizontal (X) axis in the Spectrum view.
DISPlay:SPECtrum[:MEASview<y>]:Y[:SCAle] (on page 968)	Sets or queries the vertical range of the spectrum graph.
DISPlay:SPECtrum[:MEASview<y>]:Y[:SCAle]: OFFSET (on page 969)	Sets or queries the vertical offset (the value at the top edge of the vertical axis) of the spectrum graph.
DISPlay:SPECtrum[:MEASview<y>]:Y[:SCAle]: PDIVision (on page 969)	Sets or queries the vertical scale (per division) of the spectrum graph.
DISPlay:SPECtrum[:MEASview<y>]:Y[:SCAle]:RESet (on page 970)	Resets the vertical scale of the spectrum graph to the default values
DISPLay:SPURious subgroups	**Spurious measurement**
DISPlay:SPURious:MARKer:SHOW:STATe (on page 970)	Sets or queries whether to show the readout for the selected marker.
DISPlay:SPURious:RESet:SCALe (on page 971)	Resets the horizontal and vertical scale to the default values.
DISPlay:SPURious:SCALe:LOG:STATe (on page 971)	Sets or queries whether to set the horizontal axis logarithmic.
DISPlay:SPURious:SELect:NUMBer (on page 972)	Sets or queries the spurious number.
DISPlay:SPURious:SHOW:LIMit (on page 972)	Sets or queries how to display the limits.
DISPlay:SPURious:WINDow:TRACe:GRATicule: GRID:STATe (on page 972)	Determines whether to show the graticule grid on screen.
DISPlay:SPURious:X[:SCALe]:AUTO (on page 973)	Rescales the horizontal axis automatically.

Command	Description
DISPlay:SPURious:X[:SCALe]:STARt (on page 973)	Sets or queries the minimum horizontal value of the spectrum graph.
DISPlay:SPURious:X[:SCALe]:STOP (on page 974)	Sets or queries the maximum horizontal value of the spectrum graph.
DISPlay:SPURious:Y[:SCALe] (on page 974)	Sets or queries the vertical range of the spectrum graph.
DISPlay:SPURious:Y[:SCALe]:AUTO (on page 975)	Rescales the vertical axis automatically.
DISPlay:SPURious:Y[:SCALe]:OFFSet (on page 975)	Sets or queries the vertical offset of the spectrum graph.
DISPlay:TDIagram subgroup (Option 21 only)	**Trellis diagram**
DISPlay:TDIagram[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe (on page 976)	Sets or queries whether to show or hide the graticule grid on the screen in the Trellis diagram.
DISPlay:TDIagram[:MEASview<y>]:Y[:SCALe] (on page 976)	Sets or queries the vertical scale of the Trellis diagram.
DISPlay:TDIagram[:MEASview<y>]:Y[:SCALe]:AUTO (on page 977)	Sets the vertical scale automatically.
DISPlay:TDIagram[:MEASview<y>]:Y[:SCALe]: OFFSet (on page 977)	Sets or queries the vertical offset (center point of the vertical axis).
DISPlay:TOVerview subgroup	**Time overview**
DISPlay:TOVerview[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe (on page 980)	Sets or queries whether to show the graticule grid in the time overview.
DISPlay:TOVerview[:MEASview<y>]:X[:SCALe] (on page 981)	Sets or queries the horizontal scale.
DISPlay:TOVerview[:MEASview<y>]:X[:SCALe]: AUTO (on page 982)	Sets the horizontal scale and offset automatically.
DISPlay:TOVerview[:MEASview<y>]:X[:SCALe]:OFFSet (on page 982)	Sets or queries the minimum horizontal value (left edge).
DISPlay:TOVerview[:MEASview<y>]:Y[:SCALe] (on page 983)	Sets or queries the vertical scale.
DISPlay:TOVerview[:MEASview<y>]:Y[:SCALe]: AUTO (on page 983)	Sets the vertical scale and offset automatically.
DISPlay:TOVerview[:MEASview<y>]:Y[:SCALe]:OFFSet (on page 984)	Sets or queries the vertical offset.
DISPlay:TOVerview[:MEASview<y>]:Y[:SCALe]:RESCale (on page 984)	Rescales the vertical scale.
DISPlay:WINDow subgroup	**Window display**

Command	Description
DISPlay:WINDow:ACQuisition:BANdwidth:SHOw:STAte (on page 991)	Sets or queries the state of **Show Acq Bandwidth** checkbox.
DISPlay:WINDow:ACTive:MEASurement? (on page 991)	Queries the active measurement views
DISPlay:WINDow:APPLy:SOURce:COLOr (on page 993)	Sets or queries the state of **Apply Source Color to Trace** checkbox.
DISPlay:WINDow:COLor:SCHeme (on page 994)	Sets or queries the color scheme for displaying traces and background on the screen.
DISPlay:WINDow:OPTimized:MEASurement? (on page 994)	Queries the measurement views that are optimized.
DISPlay:WLAN subgroup	**Wireless LAN measurements**
DISPlay:WLAN:CONSte[:SCALe] (on page 1013)	Sets the magnification value (zoom factor).
DISPlay:WLAN:CONSte:X:OFFSet (on page 1014)	Sets or queries the value of the offset of the horizontal axis.
DISPlay:WLAN:CONSte:Y:OFFSet (on page 1014)	Sets or queries the value of the offset of the vertical axis.
DISPlay:WLAN:CRESponse:AUTO (on page 1015)	Automatically rescales the horizontal and vertical values.
DISPlay:WLAN:CRESponse:FREQuency:AUTO (on page 1015)	Automatically rescales the horizontal subcarrier (frequency) start and stop values.
DISPlay:WLAN:CRESponse:FREQuency:STARt (on page 1016)	Specifies or queries the horizontal start value.
DISPlay:WLAN:CRESponse:FREQuency:STOP (on page 1016)	Specifies or queries the horizontal stop value.
DISPlay:WLAN:CRESponse:MAGNitude:AUTO (on page 1017)	Rescales the vertical magnitude offset (position) and scale values.
DISPlay:WLAN:CRESponse:MAGNitude:OFFSet (on page 1017)	Sets or queries the vertical offset.
DISPlay:WLAN:CRESponse:MAGNitude[:SCALe] (on page 1017)	Sets or queries the value of the vertical scale.
DISPlay:WLAN:CRESponse:MARKer:SHOW:STATe (on page 1018)	Shows or hides the marker readout.
DISPlay:WLAN:CRESponse:PHASe:AUTO (on page 1018)	Rescales the vertical phase value automatically to fit the waveform to the screen.
DISPlay:WLAN:CRESponse:PHASe:OFFSet (on page 1019)	Sets or queries the vertical offset.

Command	Description
DISPlay:WLAN:CRESponse:PHASe[:SCALe] (on page 1019)	Sets or queries the phase value.
DISPlay:WLAN:CRESponse:WINDow:SELect:PLOT (on page 1020)	Specifies or queries which graphs are displayed.
DISPlay:WLAN:CRESponse:WINDow:TRACe:GRATicule:GRID:STATe (on page 1020)	Sets or queries whether to show the graticule grid.
DISPlay:WLAN:EVM:FREQuency:AUTO (on page 1021)	Automatically rescales the horizontal subcarrier (frequency) start and stop values.
DISPlay:WLAN:EVM:FREQuency:RESet (on page 1021)	Automatically resets the horizontal subcarrier (frequency) start and stop values.
DISPlay:WLAN:EVM:FREQuency:STARt (on page 1021)	Specifies or queries the horizontal start value.
DISPlay:WLAN:EVM:FREQuency:STOP (on page 1022)	Specifies or queries the horizontal stop value.
DISPlay:WLAN:EVM:MARKer:SHOW:STATe (on page 1022)	Shows or hides the Marker readout.
DISPlay:WLAN:EVM[:SCALe]:AUTO (on page 1023)	Automatically adjusts all vertical and horizontal values.
DISPlay:WLAN:EVM:TIME:AUTO (on page 1023)	Sets the horizontal start and stop values to fit the waveform.
DISPlay:WLAN:EVM:TIME:AUTO:STATe (on page 1024)	Automatically sets the horizontal symbol scale value.
DISPlay:WLAN:EVM:TIME:RESet (on page 1024)	Resets the horizontal symbol start and stop values.
DISPlay:WLAN:EVM:TIME:STARt (on page 1025)	Sets or queries the horizontal symbol start value.
DISPlay:WLAN:EVM:TIME:STOP (on page 1025)	Sets or queries the horizontal symbol stop value
DISPlay:WLAN:EVM:WINDow:SELect:PLOT (on page 1025)	Specifies or queries which graphs are displayed in the WLAN EVM display.
DISPlay:WLAN:EVM:WINDow:TRACe:GRATicule:GRID:STATe (on page 1026)	Shows or hides the graticule grid on the screen.
DISPlay:WLAN:EVM:Y:AUTO (on page 1026)	Automatically sets the vertical scale and position values.
DISPlay:WLAN:EVM:Y:OFFSet (on page 1027)	Sets or queries the value of the offset from the center.
DISPlay:WLAN:EVM:Y:RESet (on page 1027)	Resets the vertical scale and position (offset) values.
DISPlay:WLAN:EVM:Y[:SCALe] (on page 1028)	Sets or queries the vertical scale value.
DISPlay:WLAN:FLATness:AUTO (on page 1028)	Automatically rescales the horizontal and vertical axes.

Command	Description
DISPlay:WLAN:FLATness:X:AUTO (on page 1029)	Automatically rescales the horizontal axis.
DISPlay:WLAN:FLATness:X:OFFSet (on page 1029)	Sets or queries the value of the offset of the horizontal axis from the center.
DISPlay:WLAN:FLATness:X[:SCALe] (on page 1030)	Sets or queries the horizontal scale value.
DISPlay:WLAN:FLATness:Y:AUTO (on page 1030)	Automatically rescales the vertical axis scale and position values.
DISPlay:WLAN:FLATness:Y:OFFSet (on page 1031)	Sets or queries the value of the offset from the center.
DISPlay:WLAN:FLATness:Y[:SCALe] (on page 1031)	Sets or queries the vertical scale value.
DISPlay:WLAN:MEASview:DELete (on page 1032)	Deletes the selected measurement view.
DISPlay:WLAN:MEASview:NEW (on page 1032)	Displays a new measurement view.
DISPlay:WLAN:MEASview:SELect (on page 1033)	Selects a WLAN measurement view.
DISPlay:WLAN:MERRor:FREQuency:AUTO (on page 1034)	Automatically rescales the horizontal subcarrier (frequency) start and stop values.
DISPlay:WLAN:MERRor:FREQuency:RESet (on page 1034)	Automatically resets the horizontal subcarrier (frequency) start and stop values.
DISPlay:WLAN:MERRor:FREQuency:STARt (on page 1035)	Specifies or queries the horizontal start value.
DISPlay:WLAN:MERRor:FREQuency:STOP (on page 1035)	Specifies or queries the horizontal stop value.
DISPlay:WLAN:MERRor:MARKer:SHOW:STATe (on page 1036)	Shows or hides the marker readout.
DISPlay:WLAN:MERRor[:SCALe]:AUTO (on page 1036)	Automatically adjusts all vertical and horizontal values.
DISPlay:WLAN:MERRor:TIME:AUTO (on page 1037)	Sets the horizontal start and stop values.
DISPlay:WLAN:MERRor:TIME:AUTO:STATe (on page 1037)	Automatically sets the horizontal symbol scale value.
DISPlay:WLAN:MERRor:TIME:RESet (on page 1037)	Resets the horizontal symbol start and stop values.
DISPlay:WLAN:MERRor:TIME:STARt (on page 1038)	Sets or queries the horizontal symbol start value.
DISPlay:WLAN:MERRor:TIME:STOP (on page 1038)	Sets or queries the horizontal symbol stop value.
DISPlay:WLAN:MERRor:WINDow:SELect:PLOT (on page 1039)	Specifies or queries which graphs are displayed.

Command	Description
DISPlay:WLAN:MERRor:WINDow:TRACe:GRATicule:GRID:STATe (on page 1039)	Shows or hides the graticule grid on the display.
DISPlay:WLAN:MERRor:Y:AUTO (on page 1040)	Automatically sets the vertical scale and position values.
DISPlay:WLAN:MERRor:Y:OFFSet (on page 1040)	Sets or queries the offset from the center (vertical position).
DISPlay:WLAN:MERRor:Y:RESet (on page 1041)	Resets the vertical scale and position (offset).
DISPlay:WLAN:MERRor:Y[:SCALe] (on page 1041)	Sets or queries the vertical scale value.
DISPlay:WLAN:PERRor:FREQuency:AUTO (on page 1042)	Automatically rescales the horizontal subcarrier (frequency) start and stop values.
DISPlay:WLAN:PERRor:FREQuency:RESet (on page 1042)	Automatically resets the horizontal subcarrier (frequency) start and stop values.
DISPlay:WLAN:PERRor:FREQuency:STARt (on page 1042)	Specifies or queries the horizontal start value.
DISPlay:WLAN:PERRor:FREQuency:STOP (on page 1043)	Specifies or queries the horizontal stop value.
DISPlay:WLAN:PERRor:MARKer:SHOW:STATe (on page 1043)	Show or hides the Marker readout.
DISPlay:WLAN:PERRor[:SCALe]:AUTO (on page 1044)	Automatically adjusts all vertical and horizontal settings.
DISPlay:WLAN:PERRor:TIME:AUTO (on page 1044)	Automatically sets the horizontal symbol start and stop values.
DISPlay:WLAN:PERRor:TIME:AUTO:STATe (on page 1045)	Automatically sets the horizontal symbol scale value.
DISPlay:WLAN:PERRor:TIME:RESet (on page 1045)	Resets the horizontal symbol start and stop values.
DISPlay:WLAN:PERRor:TIME:STARt (on page 1046)	Sets or queries the horizontal symbol start value.
DISPlay:WLAN:PERRor:TIME:STOP (on page 1046)	Sets or queries the horizontal symbol stop value.
DISPlay:WLAN:PERRor:WINDow:SELect:PLOT (on page 1047)	Specifies or queries which displays are displayed.
DISPlay:WLAN:PERRor:WINDow:TRACe:GRATicule:GRID:STATe (on page 1047)	Shows or hides the graticule grid on the screen.
DISPlay:WLAN:PERRor:Y:AUTO (on page 1048)	Automatically selects the vertical scale and position values.
DISPlay:WLAN:PERRor:Y:OFFSet (on page 1048)	Sets or queries the value of the offset from the center (vertical position).
DISPlay:WLAN:PERRor:Y:RESet (on page 1049)	Resets the vertical scale and position (offset) values.

Command	Description
DISPlay:WLAN:PERRor:Y[:SCALe] (on page 1049)	Sets or queries the vertical scale value.
DISPlay:WLAN:PVTime:BURSt:X[:SCALe] (on page 1049)	Sets or queries the value of the scale (width).
DISPlay:WLAN:PVTime:BURSt:X[:SCALe]:AUTO (on page 1050)	Automatically sets the starting time (position) and scale (width) values
DISPlay:WLAN:PVTime:BURSt:X[:SCALe]:OFFSet (on page 1050)	Sets or queries the starting time (position) value.
DISPlay:WLAN:PVTime:BURSt:X[:SCALe]:RESet (on page 1051)	Resets the starting time (position) and scale (width) values.
DISPlay:WLAN:PVTime:FALL:X[:SCALe] (on page 1051)	Sets or queries the scale (width) value.
DISPlay:WLAN:PVTime:FALL:X[:SCALe]:AUTO (on page 1052)	Automatically sets the starting time (position) and scale (width) values.
DISPlay:WLAN:PVTime:FALL:X[:SCALe]:OFFSet (on page 1052)	Sets or queries the value for the starting time (offset).
DISPlay:WLAN:PVTime:FALL:X[:SCALe]:RESet (on page 1053)	Resets the starting time (position) and scale (width) values.
DISPlay:WLAN:PVTime:MARKer:SHOW:STATe (on page 1053)	Shows or hides the marker readout.
DISPlay:WLAN:PVTime:RISE:X[:SCALe] (on page 1053)	Sets or queries the scale (width) value.
DISPlay:WLAN:PVTime:RISE:X[:SCALe]:AUTO (on page 1054)	Automatically sets the starting time (position) and scale (width) values.
DISPlay:WLAN:PVTime:RISE:X[:SCALe]:OFFSet (on page 1054)	Sets or queries the starting time value (offset).
DISPlay:WLAN:PVTime:RISE:X[:SCALe]:RESet (on page 1055)	Resets the starting time (position) and scale (width) values.
DISPlay:WLAN:PVTime:WINDow:SELect:PLOT (on page 1055)	Specifies or queries which view to use.
DISPlay:WLAN:PVTime:WINDow:TRACe:GRATicule:GRID:STATe (on page 1056)	Shows or hides the graticule grid on the screen.
DISPlay:WLAN:PVTime:Y[:SCALe] (on page 1056)	Sets or queries the vertical scale value.
DISPlay:WLAN:PVTime:Y[:SCALe]:AUTO (on page 1057)	Automatically selects the vertical scale and position values.

Command	Description
DISPlay:WLAN:PVTime:Y[:SCALe]:OFFSet (on page 1057)	Sets or queries the value of the vertical offset.
DISPlay:WLAN:PVTime:Y[:SCALe]:PDIVision (on page 1058)	Sets or queries the value of the vertical scale.
DISPlay:WLAN:PVTime:Y[:SCALe]:RESet (on page 1058)	Resets the vertical scale and position (offset) values.
DISPlay:WLAN:SUMMary:EVMUnits (on page 1058)	Sets or queries the EVM Units setting.
DISPlay:LTE subgroup	
DISPlay:LTE:ACLR:MARKer:SHOW:STATe (on page 768)	Determines whether to show or hide the readout for the selected marker.
DISPlay:LTE:ACLR:RESet:SCALe (on page 769)	Resets the horizontal and vertical scale to default values.
DISPlay:LTE:ACLR:SHOW:LIMit (on page 769)	Sets or queries the appearance of the limits.
DISPlay:LTE:ACLR:WINDow:TRACe:GRATicule:GRID:STATe (on page 770)	Determines whether to show or hide the graticules.
DISPlay:LTE:ACLR:WINDow:TRACe:POWer:LEVEl:STATe (on page 770)	Determines whether to show or hide the power readout.
DISPlay:LTE:ACLR:X:SCALe (on page 771)	Sets or queries the horizontal range.
DISPLay:LTE:ACLR:X:SCALe:AUTO (on page 771)	Rescales the horizontal axis automatically to fit the waveform to the screen.
DISPlay:LTE:ACLR:X:SCALe:OFFSet (on page 771)	Sets or queries the center frequency of the trace.
DISPlay:LTE:ACLR:Y:SCALe:AUTO (on page 772)	Rescales the vertical scale automatically to fit the waveform on the screen.
DISPlay:LTE:ACLR:Y:SCALe:OFFSet (on page 772)	Sets or queries the vertical offset of the trace.
DISPlay:LTE:ACLR:Y:SCALe (on page 773)	Sets or queries the vertical range of the graph.
DISPlay:LTE:CHSPectrum:MARKer:SHOW:STATe (on page 773)	Determines whether to show or hide the readout for the selected marker.
DISPlay:LTE:CHSPectrum:RESet:SCALe (on page 774)	Resets the horizontal and vertical scale to default values.
DISPlay:LTE:CHSPectrum:WINDow:TRACe:GRATicule:GRID:STATe (on page 774)	Determines whether to show or hide the graticule.
DISPlay:LTE:CHSPectrum:X:SCALe (on page 775)	Sets or queries the horizontal range.
DISPLay:LTE:CHSpectrum:X:SCALe:AUTO (on page 775)	Rescales the horizontal axis automatically to fit the waveform to the screen.

Command	Description
DISPlay:LTE:CHSPectrum:X:SCALe:OFFSet (on page 776)	Sets or queries the center frequency of the trace.
DISPlay:LTE:CHSPectrum:Y:SCALe (on page 776)	Sets or queries the vertical range of the graph.
DISPlay:LTE:CHSPectrum:Y:SCALe:AUTO (on page 776)	Rescales the vertical scale automatically to fit the waveform on the screen.
DISPlay:LTE:CHSPectrum:Y:SCALe:OFFSet (on page 777)	Sets or queries the vertical offset of the trace.
DISPlay:LTE:CONSte:SCALe (on page 777)	Sets or queries the scale (zoom).
DISPlay:LTE:CONSte:X:OFFSet (on page 778)	Sets or queries the horizontal offset.
DISPlay:LTE:CONSte:Y:OFFSet (on page 778)	Sets or queries the vertical offset.
DISPlay:LTE:MEASview:DELete (on page 779)	Deletes the selected LTE measurement display in the application.
DISPlay:LTE:MEASview:NEW (on page 779)	Opens a new LTE measurement display.
DISPlay:LTE:MEASview:SELect (on page 780)	Selects an LTE measurement display in the application. The query returns the currently selected display.
DISPlay:LTE:PVTime:MARKer:SHOW:STATe (on page 780)	Shows or hides the marker readout.
DISPlay:LTE:PVTime:WINDow:TRACe:GRATicule:GRID:STATe (on page 781)	Determines whether to show or hide the graticule.
DISPlay:LTE:PVTime:X:SCALe (on page 781)	Sets or queries the vertical range.
DISPLay:LTE:PVTime:X:SCALe:AUTO (on page 782)	Rescales the horizontal axis automatically to fit the waveform to the screen.
DISPlay:LTE:PVTime:X:SCALe:RESet (on page 782)	Resets the horizontal scale to the default value for the time measurements.
DISPlay:LTE:PVTime:Y:SCALe (on page 782)	Sets or queries the horizontal range.
DISPlay:LTE:PVTime:Y:SCALe:AUTO (on page 783)	Rescales the vertical scale automatically to fit the waveform on the screen.
DISPlay:LTE:PVTime:Y:SCALe:OFFSet (on page 783)	Sets or queries the vertical offset (top edge of the vertical axis) of the trace.
DISPlay:LTE:PVTime:Y:SCALe:PDIVision (on page 784)	Sets or queries the value of the vertical scale in dB/division.
DISPlay:LTE:PVTime:Y:SCALe:RESet (on page 784)	Resets the vertical scale and position (offset) to the default value for the time measurements.

Fetch commands

The FETCh commands retrieve the measurements from the data taken by the latest INITiate command. See the Index for commands not listed in the following table.

To perform a FETCh operation on fresh data, use the READ commands, which acquire a new input signal and fetch the measurement results from that data.

Fetch commands

Command	Description
FETCh TXGain command subgroup	**Transmission Gain measurements**
FETCh:TXGain:AVERage? (on page 1398)	Returns the mean value of the trace.
FETCh:TXGain:MAX? (on page 1399)	Returns the maximum trace value.
FETCh:TXGain:MAXLocation? (on page 1399)	Returns the frequency of the maximum trace value.
FETCh:TXGain:MIN? (on page 1399)	Returns the minimum trace value.
FETCh:TXGain:MINLocation? (on page 1400)	Returns the frequency of the minimum trace value.
FETCh:TXGain:TRACe<x>? (on page 1400)	Returns the trace data of the selected trace.
FETCh basic command subgroup	**General fetch control**
FETCh:RFIN:IQ? (on page 1356)	Returns time-domain IQ data for a specific acquisition data record.
FETCh: BLuetooth subgroup	**Bluetooth measurements**
FETCh:BIBEmissions:FTX? (on page 1085)	Returns the band number corresponding to the transmitted FTX detected in the display.
FETCh:BIBEmissions:POWer? (on page 1086)	Returns all channel power results.
FETCh:BIBEmissions:RESUlts:STATus? (on page 1086)	Returns the Pass/Fail result status.
FETCh:BLUEtooth:CONSte:FERRor? (on page 1087)	Returns the frequency error.
FETCh:BLUEtooth:CONSte:TRACe? (on page 1087)	Queries the trace data.
FETCh:BLUEtooth:EDIagram:FDEViation? (on page 1088)	Returns the frequency deviation versus time trace data with the X values.
FETCh:BLUEtooth:EDIagram:FERRor? (on page 1088)	Returns the frequency error.
FETCh:BLUEtooth:EDIagram:HORiz:SCALe? (on page 1089)	Returns the value of the horizontal scale.
FETCh:BLUEtooth:EDIagram:I? (on page 1089)	Returns the I versus time trace.

Command	Description
FETCh:BLUEtooth:EDIagram:Q? (on page 1090)	Returns the Q versus time trace.
FETCh:BLUEtooth:FDVTime:TRACe? (on page 1090)	Returns the trace data.
FETCh:BLUEtooth:FREQuency:ERROr? (on page 1091)	Returns the frequency error from the Bluetooth Constellation diagram.
FETCh:BLUEtooth:FREQuency:ERROr:TYPE? (on page 1091)	Returns the frequency error type in the Bluetooth Constellation display.
FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:F1FZero? (on page 1092)	Returns the drift in frequency offset (f1–f0—the frequency offset calculated in the first interval in the payload — the frequency offset calculated in the preamble) in the Bluetooth Frequency Deviation vs. Time display.
FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFN5? (on page 1092)	Returns the maximum drift of the frequency offset in payload intervals spaced 50 µs away in the Bluetooth Frequency Deviation vs. Time display.
FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFN5:INTErval? (on page 1093)	Returns the interval at which the maximum drift of frequency offset in a duration of 50 µs occurred.
FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFZERO? (on page 1093)	Returns the maximum drift of the frequency offset of the intervals in the payload from the preamble packet in the Bluetooth Frequency Deviation vs. Time display.
FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFZERO:INTErval? (on page 1094)	Returns the interval (n) at which the maximum drift fn-f0 occurred in the Bluetooth Frequency Deviation vs. Time display.
FETCh:BLUEtooth:FREQuency:OFFSet:MAX? (on page 1094)	Returns the maximum frequency offset in the Frequency Deviation vs. Time and the Center Frequency Offset and Drift displays.
FETCh:BLUEtooth:FREQuency:OFFSet:MAX:INTErval? (on page 1095)	Returns the maximum frequency offset interval in the Frequency Deviation vs. Time and the Center Frequency Offset and Drift displays.
FETCh:BLUEtooth:FREQuency:OFFSet:PREAmble? (on page 1095)	Returns the offset calculated in the preamble region in the Frequency Deviation vs. Time and the Center Frequency Offset and Drift displays.
FETCh:BLUEtooth:MODUlation:CHARacteristics:AVErage:F? (on page 1096)	Returns the average frequency value when the selected test pattern is Other in the Bluetooth Frequency Deviation vs. Time display.
FETCh:BLUEtooth:MODUlation:CHARacteristics:AVErage:F2F1ratio? (on page 1096)	Returns the average ratio value for the high and low frequency deviation in the Bluetooth Frequency deviation versus Time display.
FETCh:BLUEtooth:MODUlation:CHARacteristics:AVErage:FONE? (on page 1097)	Returns the average frequency deviation for the low deviation pattern in the Bluetooth Frequency deviation versus Time display.
FETCh:BLUEtooth:MODUlation:CHARacteristics:AVErage:FTWO? (on page 1097)	Returns the average frequency deviation for the high deviation pattern in the Bluetooth Frequency Deviation versus. Time display.
FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:COUNt? (on page 1098)	Returns the number of packets analyzed to obtain average drift results.

Command	Description
FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:F1FZero? (on page 1098)	Returns the drift f1–f0.
FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:F1FZero:STATus? (on page 1099)	Returns the Pass/Fail drift f1–f0 result status in the packet.
FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFN5? (on page 1099)	Returns the maximum drift of the fn–fn-5 value.
FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFN5:STATus? (on page 1100)	Returns the Pass/Fail status of the maximum fn–fn-5 drift in the packet.
FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFZero? (on page 1100)	Returns the maximum drift fn–f0.
FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFZero:STATus? (on page 1101)	Returns the Pass/Fail result status of the fn–f0 maximum drift in the packet.
FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:MAX? (on page 1105)	Returns the maximum frequency offset present in the packet.
FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:MAX:STATus? (on page 1105)	Returns the Pass or Fail result status of the frequency offset maximum in the packet.
FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:PREAmble? (on page 1106)	Returns the frequency offset value in the preamble region.
FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:PREAmble:STATus? (on page 1106)	Returns the Pass or Fail result status of the frequency offset in the preamble region.
FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:F:MAX:PERCentage:COUNt? (on page 1107)	Returns the number of packets analyzed out of 10 packets for ΔF average.
FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:FONE:MAX:PERCentage:COUNt? (on page 1107)	Returns the number of packets analyzed out of 10 packets for the ΔF1avg.
FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:FTWO:MAX:PERCentage? (on page 1108)	Returns the percentage of the ΔF2Max values that are above the specified limit.
FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:FTWO:MAX:PERCentage:COUNt? (on page 1108)	Returns the number of packets analyzed out of 10 packets for the ΔF2Max%.
FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:FTWO:MAX:PERCentage:STATus? (on page 1109)	Returns the Pass/Fail result status of the ΔF2Max% field.
FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:F? (on page 1109)	Returns the ΔFavg value.

Command	Description
FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:F2F1ratio? (on page 1110)	Returns the ratio of the ΔF2avg to ΔF1avg.
FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:F2F1ratio:STATus? (on page 1110)	Returns the Pass/Fail results status of the ΔF2avg/ΔF1avg field.
FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:FONE? (on page 1111)	Returns the ΔF1avg value.
FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:FONE:STATus? (on page 1111)	Returns the Pass/Fail results status of the ΔF1avg.
FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:FTWO? (on page 1112)	Returns the ΔF2avg value.
FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:AVErage? (on page 1113)	Returns the average output power.
FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:AVErage:STATus? (on page 1113)	Returns the Pass/Fail results status of the average output power.
FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:COUNt? (on page 1114)	Returns the number of packets analyzed out of 10 packets for thecalculation of the output power.
FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:EDR:COUNt? (on page 1114)	Returns the number of packets analyzed out of 10 packets for calculation of the output power for EDR.
FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:EDR:DPSK? (on page 1115)	Returns the power in the DPSK portion of the EDR burst.
FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:EDR:GPSK? (on page 1115)	Returns the power in the GFSK portion of the EDR burst.
FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:PEAK? (on page 1116)	Returns the peak output power.
FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:PEAK:STATus? (on page 1116)	Returns the Pass/Fail results status of the peak output power.
FETCh:BLUEtooth:SUMMary:PACKet:BDR:CRC? (on page 1116)	Returns the Basic Rate value of the CRC field.
FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:ARQN? (on page 1117)	Returns the value of ARQN field.
FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:FLOW? (on page 1117)	Returns the value of Flow field.
FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:HEC? (on page 1118)	Returns the value of HEC field.

Command	Description
FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:LT:ADDRess? (on page 1118)	Returns the LT address.
FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:SEQN? (on page 1119)	Returns the value of SEQN field.
FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:TYPE? (on page 1119)	Returns the value of Type field.
FETCh:BLUEtooth:SUMMary:PACKet:BDR:PAYLoad:LENgth? (on page 1120)	Returns the Basic Rate payload length.
FETCh:BLUEtooth:SUMMary:PACKet:BDR:PREAmble? (on page 1120)	Returns the preamble of the Basic Rate packet.
FETCh:BLUEtooth:SUMMary:PACKet:BDR:SYNC:WORD? (on page 1121)	Returns the sync word.
FETCh:BLUEtooth:SUMMary:PACKet:BDR:TYPE? (on page 1121)	Returns the packet type of the Basic Rate signal.
FETCh:BLUEtooth:SUMMary:PACKet:BLE:ACCEss:ADDRess? (on page 1122)	Returns the 32-bit Access Address.
FETCh:BLUEtooth:SUMMary:PACKet:BLE:CRC? (on page 1122)	Returns the value of the CRC field.
FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:LENGth? (on page 1123)	Returns the value of the length field.
FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:RX:ADDRess? (on page 1125)	Returns the Rx address.
FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:TX:ADDRess? (on page 1126)	Returns the Tx address.
FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:TYPE? (on page 1127)	Returns the PDU type.
FETCh:BLUEtooth:SUMMary:PACKet:BLE:PREAmble? (on page 1127)	Returns the preamble of the BLE packet.
FETCh:BLUEtooth:SUMMary:PACKet:BLE:TYPE? (on page 1128)	Returns the packet type.
FETCH:NOISe subgroup	**Noise Figure and Gain measurements**
FETCh:NOISe:GAIN:TRACe<x>:RESult? (on page 1193)	This query returns the gain value in mdB of the specified trace in the Gain display.
FETCh:NOISe:TABLe:VALue? (on page 1193)	Returns Noise Table data. A total of 7 values per frequency (28 bytes per frequency) is returned.
FETCH:OBWidth subgroup	**Bluetooth 20dB BW measurements**

Command	Description
FETCh:OBWidth:BOBW:XDBBandwidth? (on page 1220)	Returns the difference between the higher and lower frequency pointscorresponding to the value where it is X dB less from the value at the center frequency.
FETCh:OBWidth:BOBW:XDBMbandwidth:IN:BANDwidth? (on page 1220)	Returns the difference between the higher and lower frequency points corresponding to the value which it is the X dBm value measured inwards(coming from the ends to the center frequency).
FETCh:OBWidth:BOBW:XDBMbandwidth:IN:LEFT:FREQuency? (on page 1221)	Returns the frequency corresponding to the X dBm value to the left of the center frequency measured in the inward direction.
FETCh:OBWidth:BOBW:XDBMbandwidth:IN:RIGHt:FREQuency? (on page 1221)	Returns the frequency corresponding to the X dBm value to the right of the center frequency measured in the inward direction.
FETCh:OBWidth:BOBW:XDBMbandwidth:OUT:BANDwidth? (on page 1222)	Returns the difference between the higher and lower frequency points corresponding to the value which it is the X dBm value measured outwards(from the center frequency to the ends).
FETCh:OBWidth:BOBW:XDBMbandwidth:OUT:LEFT:FREQuency? (on page 1223)	Returns the frequency corresponding to the X dBm value to the left of the center frequency measured in the outward direction.
FETCh:OBWidth:BOBW:XDBMbandwidth:OUT:RIGHt:FREQuency? (on page 1223)	Returns the frequency corresponding to the X dBm value to the right of the center frequency measured in the outward direction.
FETCH:P25 subgroup	**P25 measurements**
FETCh:P25:CONSte:FERRor? (on page 1253)	Returns the frequency error.
FETCh:P25:CONSte:TRACe? (on page 1253)	Returns the display trace data.
FETCh:P25:EDIagram:FDEViation? (on page 1254)	Returns the Frequency Deviation versus Time trace data with the X value.
FETCh:P25:EDIagram:FERRor? (on page 1254)	Returns the frequency error.
FETCh:P25:EDIagram:I? (on page 1255)	Returns the I versus time trace data.
FETCh:P25:EDIagram:Q? (on page 1255)	Returns the Q versus time trace data.
FETCh:P25:PVTime:BURSt:WIDTh? (on page 1256)	Returns the value of the measured time width of the burst packet from poweron to powerdown (or the end of waveform, if that occurs before powerdown), in seconds.
FETCH:P25:SUMMary:MODUlation:MEASurement:FREQuency:ACCUracy (on page 1263)	Returns the Operational Frequency Accuracy measurement results.
FETCh:P25:SUMMary:MODUlation:MEASurement:FREQuency:ACCUracy:STATus (on page 1263)	Returns the Operational Frequency Accuracy measurement status.
FETCH:P25:SUMMary:MODUlation:MEASurement:FREQuency:DEVIation:NEGPeak (on page 1264)	Returns the Frequency Deviation, negative peak, measurement results.

Command	Description
FETCh:P25:SUMMary:MODUlation:MEASurement:FREQuency:DEVIation:NEGPeak:STATus (on page 1264)	Returns the Status of Frequency Deviation, negative peak measurement status.
FETCH:P25:SUMMary:MODUlation:MEASurement:FREQuency:DEVIation:POSPeak (on page 1265)	Returns the status of Frequency Deviation, positive peak resutls.
FETCh:P25:SUMMary:MODUlation:MEASurement:FREQuency:.DEVIation:POSPeak:STATus (on page 1265)	Returns the status of Frequency Deviation, positive peak status.
FETCH:P25:SUMMary:MODUlation:MEASurement:FREQuency:ERRor (on page 1266)	Returns the Frequency Error measurement results.
FETCH:P25:SUMMary:MODUlation:MEASurement:SRA? (on page 1266)	Returns the Symbol Rate Accuracy measurement results.
FETCh:P25:SUMMary:MODUlation:MEASurement:SRA:STATus (on page 1267)	Returns the Symbol Rate Accuracy measurement status.
FETCh:P25:SUMMary:POWEr:MEASurement:HCPM:MAXOFF:STATus? (on page 1267)	Returns HCPM Pmax-off measurement status.
FETCh:P25:SUMMary:POWEr:MEASurement:HCPM:MAXON:STATus? (on page 1268)	Returns Status of HCPM Pmax-on measurement status.
FETCh:P25:SUMMary:POWEr:MEASurement:HCPM:MAXSs:STATus (on page 1269)	Returns HCPM Pss-max measurement status.
FETCh:P25:SUMMary:POWEr:MEASurement:HCPM:MINImum:STATus? (on page 1270)	Returns status of HCPM Pss-min measurement status based on the limit.
FETCh:P25:SUMMary:POWEr:MEASurement:HCPM:OFFSlot:STATus? (on page 1271)	Returns status of HCPM Offslot measurement based on the limit.
FETCh:P25:SUMMary:POWEr:MEASurement:HCPM:PEAK:ACPR:HI (on page 1271)	Returns P_ACP_HI value.
FETCh:P25:SUMMary:POWEr:MEASurement:HCPM:PEAK:ACPR:LOW (on page 1272)	Returns P_ACP_LOW value.
FETCh:P25:SUMMary:POWEr:MEASurement:HCPM:PEAK:ACPR:MIN (on page 1272)	Returns Min Pk ACPR value.
FETCh:P25:SUMMary:POWEr:MEASurement:HCPM:PEAK:ACPR:MIN:STATus (on page 1273)	Returns status of Min Pk ACPR value based on the limit.
FETCh:P25:SUMMary:POWEr:MEASurement:RF (on page 1273)	Returns RF measurement results.
FETCh:P25:SUMMary:POWEr:MEASurement:RF:STATus (on page 1273)	Returns Status of RF measurement status.

Command	Description
FETCH:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ONE:ACQ:COUNT? (on page 1274)	Returns the HCPM Time alignment's t_error_1 acquisition count over which the t_error_1 average is computed.
FETCh:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ONE:STATus (on page 1275)	Returns status of HCPM Time alignment's t_error_1 measurement results based on the limit.
FETCh:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ZERO:STATus? (on page 1275)	Returns status of HCPM Time alignment's t_error_0 measurement results.
FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ENCOder:BI:STATus? (on page 1282)	Returns status of Average Transmitter Power Encoder Attack Time,busy/idle, measurement status.
FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ENCOder:STATus? (on page 1280), FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ENCOder:STATus? (on page 1283)	Returns status of Average Transmitter Encoder Attack Time measurement results based on the limit.
FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ENCOder? (on page 1283)	Returns the Average Transmitter Encoder Attack Time measurement results.
FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr:BI:STATus (on page 1285)	Returns status of Average Transmitter Power Attack Time, busy/idle,measurement results based on the limit.
FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr:BI? (on page 1281), FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr:BI? (on page 1284)	Returns the Average Transmitter Power Attack Time, busy/idle,measurement results.
FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr:STATus (on page 1286)	Returns status of Average Transmitter Power Attack Time measurement results.
FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr? (on page 1286)	Returns the Average Transmitter Power Attack Time measurement results.
FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy:ACQ:COUNT? (on page 1287)	Returns the Throughput delay acquisition counts over which the Throughput delay average is computed.
FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy:STATus (on page 1288)	Returns status of Throughput delay measurement results based on the limit.
FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy? (on page 1288)	Returns the Throughput delay measurement results.
FETCH:P25:SUMMary:POWEr:MEASurement:HCPM:MAXOFF? (on page 1267)	Returns the HCPM Pmax-off measurement results.

Command	Description
FETCH:P25:SUMMary:POWEr:MEASurement:HCPM:MAXON? (on page 1268)	Returns the HCPM Pmax-on measurement results.
FETCH:P25:SUMMary:POWEr:MEASurement:HCPM:MAXSS? (on page 1269)	Returns the HCPM Pss-max measurement results.
FETCH:P25:SUMMary:POWEr:MEASurement:HCPM:MINImum? (on page 1270)	Returns the HCPM Pss-min measurement results.
FETCH:P25:SUMMary:POWEr:MEASurement:HCPM:OFFSlot? (on page 1270)	Returns the HCPM Offslot power results.
FETCH:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ONE? (on page 1277)	Returns the HCPM Time alignment's t_error_1 measurement results.
FETCh:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ONE:.ACQ:COUNT? (on page 1277)	Returns the HCPM Time alignment's t_error_1 acquisition count over which the t_error_1 average is computed.
FETCH:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ZERO? (on page 1278)	Returns the HCPM Time alignment's t_error_0 measurement results.
FETCh:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ZERO:ACQ:COUNT (on page 1276)	Returns the HCPM Time alignment's t_error_0 acquisition count over which the t_error_0 average is done.
FETCH:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TOBSync? (on page 1278)	Returns the HCPM Time alignment's t_obsync measurement results.
FETCh:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TOBSync:ACQ:COUNT (on page 1279)	Returns the HCPM Time alignment's t_obsync acquisition counts over which the t_obsync average is computed.
FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ACQ:COUNT (on page 1281)	Returns the Average Transmitter Power Attack acquisition count over which the t_obsync average is computed.
FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ACQ:COUNT:BI? (on page 1281)	Returns the Average Transmitter Power Attack, busy/idle, acquisition count over which the Busy/Idle Attack time average is computed.
FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ENCOder? (on page 1283)	Returns the Average Transmitter Encoder Attack Time measurement results.
FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ENCOder:BI? (on page 1284)	Returns the Average Transmitter Encoder Attack Time, busy/idle,measurement results.
FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr (on page 1284)	Returns the Average Transmitter Power Attack Time measurement results.

Command	Description
FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr:BI (on page 1285)	Returns the Average Transmitter Power Attack Time, busy or idle,measurement results.
FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy (on page 1288)	Returns the Throughput delay measurement results.
FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy:ACQ:COUNT (on page 1287)	Returns the Throughput delay acquisition counts over which the Throughput delay average is computed.
FETCh:RFIN subgroup	**RF acquisition data**
FETCh:RFIN:IQ:HEADer? (on page 1357)	Returns the header information for a specific acquisition data record.
FETCh:RFIN:IQ:SCALe? (on page 1357)	Returns the scaling factor contained in the .tiq file header.
FETCh:RFIN:RECord:IDS? (on page 1359)	Returns the beginning and end ID numbers of acquisition data.
FETCh:ACPower subgroup	**Channel power and ACPR measurement**
FETCh:ACPower? (on page 1060)	Returns the ACPR measurement results.
FETCh:ACPower:CHANnel:POWer? (on page 1061)	Returns the average power of the main channel.
FETCh:ACPower:SPECtrum? (on page 1061)	Returns spectrum trace data of the ACPR measurement.
FETCh:FM:PNEGative? (on page 1159)	Returns the negative peak frequency deviation (–Pk).
FETCh:FM:PPOSitive? (on page 1160)	Returns the positive peak frequency deviation (+Pk).
FETCh:FM:PTPeak? (on page 1160)	Returns the peak-peak frequency deviation (Pk-Pk).
FETCh:FM:RESult? (on page 1161)	Returns the FM measurement results.
FETCh:FM:RMS? (on page 1161)	Returns the RMS frequency deviation.
FETCh:PM:PNEGative? (on page 1296)	Returns the negative peak phase deviation (–Pk).
FETCh:PM:PPOSitive? (on page 1297)	Returns the negative peak phase deviation (–Pk).
FETCh:PM:PTPeak? (on page 1297)	Returns the peak-peak phase deviation (Pk-Pk).
FETCh:PM:RESult? (on page 1297)	Returns the PM measurement results.
FETCh:PM:RMS? (on page 1298)	Returns the PM measurement results.
FETCh:AUDio subgroup	**Audio measurement**
FETCh:AUDio:FERRor? (on page 1064)	Returns the carrier frequency error in the audio measurement.
FETCh:AUDio:FREQuency? (on page 1065)	Returns the audio frequency.

Command	Description
FETCh:AUDio:HARMonic:COUNt? (on page 1065)	Returns the number of harmonics in the audio spectrum measurement.
FETCh:AUDio:HARMonic<x>:AMPLitude? (on page 1066)	Returns the amplitude of the specified harmonic in the audio spectrum measurement.
FETCh:AUDio:HARMonic<x>:FREQuency? (on page 1066)	Returns the frequency of the specified harmonic in the audio spectrum measurement.
FETCh:AUDio:HNOise? (on page 1067)	Returns the difference between the current RMS modulation value and the reference value.
FETCh:AUDio:HNREFerence? (on page 1067)	Returns the Hum and Noise RMS modulation reference value.
FETCh:AUDio:HPTPeak? (on page 1068)	Returns the half peak-to-peak modulation excursion.
FETCh:AUDio:MODDist? (on page 1069)	Returns the modulation distortion for the audio measurement.
FETCh:AUDio:NHARmonic:COUNt? (on page 1069)	Returns the number of non-harmonics in the audio spectrum measurement.
FETCh:AUDio:NHARmonic<x>:AMPLitude? (on page 1070)	Returns the amplitude of the specified non-harmonic in the audio spectrum measurement.
FETCh:AUDio:NHARmonic<x>:FREQuency? (on page 1070)	Returns the frequency of the specified non-harmonic in the audio spectrum measurement.
FETCh:AUDio:PNEGative? (on page 1071)	Returns the minus peak modulation excursion.
FETCh:AUDio:POWer? (on page 1071)	Returns the carrier power or signal power depending the signal type.
FETCh:AUDio:PPOSitive? (on page 1072)	Returns the positive peak modulation excursion.
FETCh:AUDio:RMS? (on page 1072)	Returns the RMS modulation excursion for the audio measurement.
FETCh:AUDio:SINad? (on page 1073)	Returns the signal-to-noise and distortion for the audio measurement.
FETCh:AUDio:SNOise? (on page 1073)	Returns the signal level to noise level with the harmonic distortion and non-harmonic distortion components removed.
FETCh:AUDio:SPECtrum:TRACe<x>? (on page 1074)	Returns the audio spectrum trace data for the audio measurement.
FETCh:AUDio:THDist? (on page 1074)	Returns the percentage of the total harmonic distortion.
FETCh:AUDio:THDist:DB? (on page 1075)	Returns the total harmonic distortion in dB in the audio measurement.
FETCh:AUDio:TNHDist? (on page 1075)	Returns the percentage of the total harmonic distortion in the audio measurement.
FETCh:AUDio:TNHDist:DB? (on page 1076)	Returns the total non-harmonic distortion in dB in the audio measurement.

Command	Description
FETCh:AUDio:TNHDist? (on page 1075)	Returns the percentage of the total non-harmonic distortion in the audio measurement.
FETCh:AVTime subgroup	**Amplitude versus Time measurement**
FETCh:AVTime[:MEASview<y>]:AVERage? (on page 1076)	Returns the RMS value.
FETCh:AVTime[:MEASview<y>]: {FIRSt|SECond|THIRd|FOURth}? (on page 1077)	Returns the trace data.
FETCh:AVTime[:MEASview<y>]:MAXimum? (on page 1082)	Returns the maximum value.
FETCh:AVTime[:MEASview<y>]:MAXLocation? (on page 1082)	Returns the time at the maximum.
FETCh:AVTime[:MEASview<y>]:MINimum? (on page 1083)	Returns the minimum value.
FETCh:AVTime[:MEASview<y>]:MINLocation? (on page 1083)	Returns the time at the minimum.
FETCh:AVTime[:MEASview<y>]:OGRAm? (Query only) (on page 1084)	Returns trace data of a line in the ogram.
FETCh:AVTime[:MEASview<y>]:RESult? (on page 1085)	Returns the measurement results.
FETCh:AVTime[:MEASview<y>]:TRACe:BITMap? (Query Only) (on page 1080)	Returns trace waveform data of the bitmap trace in the Amplitude versus Time measurement.
FETCh:CCDF subgroup	**CCDF measurement**
FETCh:CCDF? (on page 1128)	Returns the CCDF measurement results.
FETCh:CCDF:{FIRSt|SECond|THIRd}:X? (on page 1129)	Returns the horizontal values of the specified trace.
FETCh:CCDF:{FIRSt|SECond|THIRd}:XY? (on page 1129)	Returns the horizontal and vertical value pairs of the specified trace.
FETCh:CCDF:{FIRSt|SECond|THIRd}[:Y]? (on page 1130)	Returns the vertical values of the specified trace.
FETCh:CONSte subgroup (Option 21 only)	**Constellation measurement**
FETCh:CONSte[:MEASview<x>]:FERRor? (on page 1131)	Returns the frequency error in Hz. The frequency error is the difference between the measured carrier frequency of the signal and the user-selected center frequency of the analyzer.
FETCh:CONSte[:MEASview<x>]:RESults? (on page 1131)	Returns the constellation measurement results of EVM RMS, peak and location displayed on the bottom of the screen.

Command	Description
FETCh:CONSte[:MEASview<x>]:TRACe? (on page 1132)	Returns the constellation trace data.
FETCh:DDEMod subgroup (Option 21 only)	**General purpose digital modulation measurements**
FETCh:DDEMod[:MEASview<y>]:STABle? (on page 1133)	Returns the symbol table data.
FETCh:DDEMod:SYNCh:WORD:LENGth? (on page 1133)	Returns the length of the synch word in the symbol table.
FETCh:DDEMod:SYNCh:WORD:POSition? (on page 1134)	Returns the position of the synch word in the symbol table.
FETCh:DIQVtime subgroup (Option 21 only)	**Demodulated I&Q versus Time measurement**
FETCh:DIQVtime[:MEASview<y>]:FERRor? (on page 1134)	Returns the frequency error.
FETCh:DIQVtime[:MEASview<y>]:I? (on page 1135)	Returns the I versus Time trace data.
FETCh:DIQVtime[:MEASview<y>]:Q? (on page 1136)	Returns the Q versus Time trace data.
FETCh:DPX subgroup	**DPX spectrum measurement**
FETCh:DPX:DDENsity? (on page 1136)	Returns the calculated density for the selected region in the DPXogram measurement.
FETCh:DPX:DGRam:LINE:COUNt? (on page 1137)	Returns the number of lines in the DPXogram measurement.
FETCh:DPX:DGRam:TIME[:SCALe]:OFFSet? (on page 1137)	Returns the time scale offset in the DPXogram measurement.
FETCh:DPX:RESults:TRACe<x>? (on page 1138)	Returns waveform data in the DPX spectrum measurement.
FETCh:DPX:TDM:FREQuency:TXBer:BER? (on page 1138)	Returns the Tx BER measurement BER results.
FETCh:DPX:TDM:FREQuency:TXBer:BITS? (on page 1139)	Returns the number of bits counted in the active Tx BER measurement session.
FETCh:DPX:TDM:FREQuency:TXBer:ERRors? (on page 1139)	Returns the number of errors counted in the active Tx BER measurement session.
FETCh:DPX:TDM:FREQuency:TXBer:STATe? (on page 1140)	Returns the state of the Tx BER measurement.
FETCh:DPX:TRACe:AVERage? (on page 1140)	Returns waveform data of the average trace in the DPX spectrum measurement.
FETCh:DPX:TRACe:BITMap? (on page 1141)	Returns trace waveform data of the bitmap trace.

Command	Description
FETCh:DPX:TRACe:MATH? (on page 1141)	Returns waveform data of the math trace.
FETCh:DPX:TRACe:MAXimum? (on page 1142)	Returns waveform data of the maximum trace.
FETCh:DPX:TRACe:MINimum? (on page 1142)	Returns waveform data of the minimum trace.
FETCh:EDIagram subgroup (Option 21 only)	**Eye diagram**
FETCh:EDIagram[:MEASview<y>]:FDEViation? (on page 1143)	Returns the frequency deviation versus Time trace data with the X values.
FETCh:EDIagram[:MEASview<y>]:FERRor? (on page 1144)	Returns the frequency error in the eye diagram measurement.
FETCh:EDIagram[:MEASview<y>]:I? (on page 1144)	Returns the I versus Time trace data in the eye diagram measurement.
FETCh:EDIagram[:MEASview<y>]:Q? (on page 1145)	Returns the Q versus Time trace data in the eye diagram measurement.
FETCh:EVM subgroup (Option 21 only)	**EVM versus Time measurement**
FETCh:EVM[:MEASview<y>]:FERRor? (on page 1154)	Returns the frequency error in the EVM versus Time measurement.
FETCh:EVM[:MEASview<y>]:PEAK? (on page 1155)	Returns the peak value in the EVM versus Time measurement.
FETCh:EVM[:MEASview<y>]:PINDex? (on page 1155)	Returns the time at the EVM peak.
FETCh:EVM[:MEASview<y>]:RMS? (on page 1156)	Returns the RMS (Root-Mean-Square) value in the EVM versus Time measurement.
FETCh:EVM[:MEASview<y>]:TRACe? (on page 1158)	Returns the EVM versus Time trace data.
FETCh:FDVTime subgroup (Option 21 only)	**Frequency deviation versus Time measurement**
FETCh:FDVTime[:MEASview<y>]:FERRor? (on page 1157)	Returns the frequency error in the Frequency deviation versus Time measurement.
FETCh:FDVTime[:MEASview<y>]:TRACe? (on page 1158)	Returns the Frequency deviation versus Time trace data.
FETCh:{FSETtling|PSETtling} subgroup (Option 12 only)	**Frequency and Phase Settling measurements**
FETCh:FSETtling:ERRor? (on page 1168)	Returns the settled error in Hz in Frequency Settling.
FETCh:{FSETtling|PSETtling}:FTTime? (on page 1162)	Returns the settling time from the trigger position in seconds.

Command	Description
FETCh:{FSETtling|PSETtling}:MASK[:PASS]? (on page 1162)	Returns whether the input signal passes the mask test.
FETCh:{FSETtling|PSETtling}:SETTled:FREQuency? (on page 1163)	Returns the frequency at which the signal is considered settled.
FETCh:{FSETtling|PSETtling}:SETTled[:PASS]? (on page 1163)	Returns whether the input signal is settled with the tolerance range.
FETCh:{FSETtling|PSETtling}:SETTled:TIME? (on page 1163)	Returns the settled time in seconds.
FETCh:{FSETtling|PSETtling}:SLMSd[:PASS]? (on page 1164)	Returns whether the input signal is settled within the specified tolerance and the signal is settled longer than the Minimum Settled Duration.
FETCh:{FSETtling|PSETtling}:STARt:TIME? (on page 1164)	Returns the start time for the measurement in seconds.
FETCh:{FSETtling|PSETtling}:TIME? (on page 1165)	Returns the settling time in seconds. The settling time is measured from the start time.
FETCh:{FSETtling|PSETtling}:TRACe<x>:X? (on page 1165)	Returns the settling time in seconds. The settling time is measured from the start time.
FETCh:{FSETtling|PSETtling}:TRACe<x>:XY? (on page 1166)	Returns the time and frequency or phase value pairs of the specified trace.
FETCh:{FSETtling|PSETtling}:TRACe<x>[:Y]? (on page 1167)	Returns the frequency or phase values of the specified trace.
FETCh:{FSETtling|PSETtling}:TRIGger:TIME? (on page 1167)	Returns the time when the trigger occurred in seconds.
FETCh:{FSETtling|PSETtling}:VALue? (on page 1168)	Returns the settled value in Hz for Frequency Settling and in degrees for Phase Settling.
FETCh:FVTime subgroup	**Frequency versus Time measurement**
FETCh:FVTime[:MEASview<y>]? (on page 1169)	Returns the Frequency versus Time trace data.
FETCh:FVTime[:MEASview<y>]:MAXimum? (on page 1169)	Returns the maximum value.
FETCh:FVTime[:MEASview<y>]:MAXLocation? (on page 1170)	Returns the time at which the frequency drift is maximum.
FETCh:FVTime[:MEASview<y>]:MINimum? (on page 1170)	Returns the minimum value.
FETCh:FVTime[:MEASview<y>]:MINLocation? (on page 1171)	Returns the time at which the frequency drift is minimum.

Command	Description
FETCh:FVTime[:MEASview<y>]:RESult? (on page 1172)	Returns the measurement results.
FETCh:FVTime[:MEASview<y>]:TRACe:BITMap? (Query Only) (on page 1173)	Returns trace waveform data of the bitmap trace in the Frequency versus Time measurement.
FETCh:IQVTime subgroup	**RF I&Q versus Time measurement**
FETCh:IQVTime[:MEASview<y>]:I? (on page 1173)	Returns the I versus Time trace data.
FETCh:IQVTime[:MEASview<y>]:MAXimum? (on page 1174)	Returns the maximum value.
FETCh:IQVTime[:MEASview<y>]:MAXLocation? (on page 1174)	Returns the time at which the I or Q level is maximum.
FETCh:IQVTime[:MEASview<y>]:MINimum? (on page 1175)	Returns the minimum value.
FETCh:IQVTime[:MEASview<y>]:MINLocation? (on page 1175)	Returns the time at which the I or Q level is minimum.
FETCh:IQVTime[:MEASview<y>]:Q? (on page 1176)	Returns the Q versus Time trace data.
FETCh:IQVTime[:MEASview<y>]:RESult? (on page 1176)	Returns the measurement results.
FETCh:MCPower subgroup	**MCPR measurement**
FETCh:MCPower:ADJacent:CHANnels? (on page 1188)	Returns the power of adjacent channels.
FETCh:MCPower:CHANnel:POWer? (on page 1188)	Returns the reference power.
FETCh:MCPower:MAIN:CHANnels? (on page 1189)	Returns the power of main channels.
FETCh:MCPower:SPECtrum? (on page 1190)	Returns spectrum trace data.
FETCh:MERRor subgroup (Option 21 only)	**Magnitude error versus Time measurement**
FETCh:MERRor[:MEASview<y>]:FERRor? (on page 1190)	Returns the frequency error in the Magnitude error versus Time measurement.
FETCh:MERRor[:MEASview<y>]:PEAK? (on page 1191)	Returns the peak value in the Magnitude error versus Time measurement.
FETCh:MERRor[:MEASview<y>]:PINDex? (on page 1191)	Returns the time at the Magnitude error peak.
FETCh:MERRor[:MEASview<y>]:RMS? (on page 1192)	Returns the RMS value in the Magnitude error versus Time measurement.

Command	Description
FETCh:MERRor[:MEASview<y>]:TRACe? (on page 1192)	Returns the Magnitude error versus Time trace data.
FETCh:OBWidth subgroup	**Occupied Bandwidth measurement**
FETCh:OBWidth:BOBW:FREQuency:ERRor (on page 1218)	Returns the frequency error.
FETCh:OBWidth:BOBW:RESUlts:STATus (on page 1219)	Returns the test result as Pass or Fail.
FETCh:OBWidth:BOBW:XDBB:POWer (on page 1219)	Returns the dB BW reference power.
FETCh:OBWidth:FREQuency:ERRor? (on page 1224)	Returns the frequency error.
FETCh:OBWidth:OBWidth:BANDwidth? (on page 1224)	Returns the occupied bandwidth.
FETCh:OBWidth:OBWidth:LEFT:FREQuency? (on page 1225)	Returns the left (lower) frequency of the occupied bandwidth.
FETCh:OBWidth:OBWidth:LEFT:LEVel? (on page 1225)	Returns the level at the left frequency of the occupied bandwidth.
FETCh:OBWidth:OBWidth:POWer? (on page 1226)	Returns the reference power in the Occupied Bandwidth measurement.
FETCh:OBWidth:OBWidth:RIGHt:FREQuency? (on page 1226)	Returns the right (higher) frequency of the occupied bandwidth.
FETCh:OBWidth:OBWidth:RIGHt:LEVel? (on page 1227)	Returns the level at the right frequency of the occupied bandwidth.
FETCh:OBWidth:SPECtrum? (on page 1227)	Returns spectrum trace data of the Occupied Bandwidth measurement.
FETCh:OBWidth:XDBBandwidth:BANDwidth? (on page 1228)	Returns the x dB bandwidth.
FETCh:OBWidth:XDBBandwidth:LEFT:FREQuency? (on page 1228)	Returns the left (lower) frequency of the x dB bandwidth.
FETCh:OBWidth:XDBBandwidth:LEFT:LEVel? (on page 1229)	Returns the level at the left frequency of the x dB bandwidth.
FETCh:OBWidth:XDBBandwidth:POWer? (on page 1229)	Returns the reference power in the x dB bandwidth measurement.
FETCh:OBWidth:XDBBandwidth:RIGHt:FREQuency? (on page 1230)	Returns the right (higher) frequency of the x dB bandwidth.
FETCh:OBWidth:XDBBandwidth:RIGHt:LEVel? (on page 1230)	Returns the level at the right frequency of the x dB bandwidth.

Command	Description
FETCh:OFDM subgroup	**Orthogonal Frequency Division Multiplexing measurement**
FETCh:OFDM:APOWer? (on page 1231)	Returns the average power.
FETCh:OFDM:APOWer:PEAK? (on page 1231)	Returns the peak-to-average power.
FETCh:OFDM:CONSte:MAGNitude? (on page 1232)	Returns the constellation magnitude data.
FETCh:OFDM:CONSte:PHASe? (on page 1232)	Returns the constellation phase values.
FETCh:OFDM:CONSte:TYPE? (on page 1233)	Returns the constellation context value.
FETCh:OFDM:CONSte:VALue? (on page 1234)	Returns the constellation value.
FETCh:OFDM:CPE? (on page 1234)	Returns the Common Pilot Error magnitude.
FETCh:OFDM:CRESponse:MAGNitude? (on page 1235)	Returns the channel response magnitude data.
FETCh:OFDM:CRESponse:PHASe? (on page 1235)	Returns the channel response phase data.
FETCh:OFDM:EVM:PEAK:DECibel:ALL? (on page 1236)	Returns the peak EVM data for all subcarriers.
FETCh:OFDM:EVM:PEAK:DECibel:DATA? (on page 1236)	Returns the peak EVM data for the data subcarriers.
FETCh:OFDM:EVM:PEAK:DECibel:PILots? (on page 1237)	Returns the peak EVM data for the pilot subcarriers.
FETCh:OFDM:EVM:PEAK:PERCent:ALL? (on page 1237)	Returns the peak EVM data for all subcarriers.
FETCh:OFDM:EVM:PEAK:PERCent:DATA? (on page 1238)	Returns the peak EVM data for the data subcarriers.
FETCh:OFDM:EVM:PEAK:PERCent:PILots? (on page 1238)	Returns the peak EVM data for the pilot subcarriers.
FETCh:OFDM:EVM:PEAK:SCARrier:ALL? (on page 1239)	Returns the peak EVM data for all subcarriers.
FETCh:OFDM:EVM:PEAK:SCARrier:DATA? (on page 1239)	Returns the peak EVM data for all data subcarriers.
FETCh:OFDM:EVM:PEAK:SCARrier:PILots? (on page 1240)	Returns the peak EVM data for the pilot subcarriers.
FETCh:OFDM:EVM:PEAK:SYMBol:ALL? (on page 1240)	Returns the peak EVM data for all subcarriers.

Command	Description
FETCh:OFDM:EVM:PEAK:SYMBol:DATA? (on page 1240)	Returns the peak EVM data for the data subcarriers.
FETCh:OFDM:EVM:PEAK:SYMBol:PILots? (on page 1241)	Returns the peak EVM data for the pilot subcarriers.
FETCh:OFDM:EVM:RMS:DECibel:ALL? (on page 1241)	Returns the RMS EVM data for all subcarriers
FETCh:OFDM:EVM:RMS:DECibel:DATA? (on page 1242)	Returns the RMS EVM data for the data subcarriers.
FETCh:OFDM:EVM:RMS:DECibel:PILots? (on page 1242)	Returns the RMS EVM data for the pilot subcarriers.
FETCh:OFDM:EVM:RMS:PERCent:ALL? (on page 1243)	Returns the peak RMS data for all subcarriers
FETCh:OFDM:EVM:RMS:PERCent:DATA? (on page 1243)	Returns the peak RMS data for the data subcarriers.
FETCh:OFDM:EVM:RMS:PERCent:PILots? (on page 1244)	Returns the peak RMS data for the pilot subcarriers.
FETCh:OFDM:EVM:TRACe<x>? (on page 1244)	Returns the EVM trace data.
FETCh:OFDM:FERRor? (on page 1245)	Returns the Frequency error reading.
FETCh:OFDM:FLATness:PASS? (on page 1245)	Verifies the average power levels of the subcarriers remain within the limits defined for a particular standard.
FETCh:OFDM:FLATness:RESult? (on page 1246)	Returns the summary results of each segment.
FETCh:OFDM:FLATness:TRACe<x>? (on page 1246)	Returns the OFDM Spectral Flatness trace data.
FETCh:OFDM:GAIN:IMBalance? (on page 1247)	Returns the gain imbalance.
FETCh:OFDM:IQ:ORIGin:OFFSet? (on page 1247)	Returns the IQ origin offset.
FETCh:OFDM:MERRor:TRACe<x>? (on page 1248)	Returns the magnitude error trace data.
FETCh:OFDM:PACKet:DIRection? (on page 1248)	Returns the direction of the packet.
FETCh:OFDM:PERRor:TRACe<x>? (on page 1249)	Returns the phase error trace data.
FETCh:OFDM:POWer:TRACe<x>? (on page 1250)	Returns the power trace data.
FETCh:OFDM:QUADrature:OFFSet? (on page 1250)	Returns the quadrature offset.

Command	Description
FETCh:OFDM:SCARriers? (on page 1251)	Returns the number of subcarriers.
FETCh:OFDM:SCARriers:SPACing? (on page 1251)	Returns the subcarrier spacing.
FETCh:OFDM:STABle:VALUe? (on page 1252)	Returns the number of bytes in the symbol table.
FETCh:OFDM:SYMBol:CERRor? (on page 1252)	Returns the symbol clock error.
FETCh:OFDM:SYMBol:COUNt? (on page 1253)	Returns the number of symbols.
FETCh:PERRor subgroup (Option 21 only)	**Phase error versus Time measurement**
FETCh:PERRor[:MEASview<y>]:FERRor? (on page 1289)	Returns the frequency error in the Phase error versus Time measurement.
FETCh:PERRor[:MEASview<y>]:PEAK? (on page 1289)	Returns the peak value in the Phase error versus Time measurement.
FETCh:PERRor[:MEASview<y>]:PINDex? (on page 1290)	Returns the time at the phase error peak.
FETCh:PERRor[:MEASview<y>]:RMS? (on page 1290)	Returns the RMS (Root-Mean-Square) value in the Phase error versus Time measurement.
FETCh:PERRor[:MEASview<y>]:TRACe? (on page 1291)	Returns the Phase error versus Time trace data.
FETCh:PHVTime subgroup	**Phase versus Time measurement**
FETCh:PHVTime[:MEASview<y>]? (on page 1292)	Returns the Phase versus Time trace data.
FETCh:PHVTime[:MEASview<y>]:MAXimum? (on page 1292)	Returns the maximum value.
FETCh:PHVTime[:MEASview<y>]:MAXLocation? (on page 1293)	Returns the time at which the phase is maximum.
FETCh:PHVTime[:MEASview<y>]:MINimum? (on page 1293)	Returns the minimum value.
FETCh:PHVTime[:MEASview<y>]:MINLocation? (on page 1294)	Returns the time at which the phase is minimum.
FETCh:PHVTime[:MEASview<y>]:OGRAm? (Query only) (on page 1294)	Returns trace data of a line in the ogram.
FETCh:PHVTime[:MEASview<y>]:RESult? (on page 1295)	Returns the results.
FETCh:PHVTime[:MEASview<y>]:TRACe:BITMap? (Query Only) (on page 1296)	Returns trace waveform data of the bitmap trace in the Phase versus Time measurement.
FETCh:PNOise subgroup (Option 11 only)	**Phase noise measurements**

Command	Description
FETCh:PNOise:ALL? (on page 1298)	Returns all the measurement results.
FETCh:PNOise:CARRier:FERRor? (on page 1299)	Returns the carrier frequency error.
FETCh:PNOise:CARRier:POWer? (on page 1300)	Returns the carrier power.
FETCh:PNOise:RESidual:FM? (on page 1300)	Returns the residual FM.
FETCh:PNOise:RMS:PNOise? (on page 1301)	Returns the RMS phase noise.
FETCh:PNOise:SPECtrum<x>:X? (on page 1301)	Returns the frequencies of the specified trace.
FETCh:PNOise:SPECtrum<x>:XY? (on page 1302)	Returns the frequency and phase noise pairs of the specified trace.
FETCh:PNOise:SPECtrum<x>[:Y]? (on page 1303)	Returns the phase noise values of the specified trace.
FETCh:PNOise:JITTer? (on page 1300)	Returns the jitter.
FETCh:PULSe subgroup	**Pulsed RF measurements**
FETCh:PULSe[:RESult][:MEASview<y>]:ATX? (on page 1313)	Returns the average transmitted power in the results table.
FETCh:PULSe[:RESult][:MEASview<y>]:AVERage? (on page 1313)	Returns the average on power in the results table.
FETCh:PULSe[:RESult][:MEASview<y>]:DRODb? (on page 1314)	Returns the droop in dB in the pulse table.
FETCh:PULSe[:RESult][:MEASview<y>]:DROop? (on page 1314)	Returns the droop in the pulse table
FETCh:PULSe[:RESult][:MEASview<y>]:DUTPct? (on page 1315)	Returns the duty factor (%) in the results table.
FETCh:PULSe[:RESult][:MEASview<y>]:DUTRatio? (on page 1316)	Returns the duty factor (ratio) in the results table.
FETCh:PULSe[:RESult][:MEASview<y>]:FALL? (on page 1317)	Returns the fall time in the results table.
FETCh:PULSe[:RESult][:MEASview<y>]:FDELta? (on page 1317)	Returns the delta frequency in the results table.
FETCh:PULSe[:RESult][:MEASview<y>]:FRDeviation? (on page 1318)	Returns the frequency deviation in the results table.
FETCh:PULSe[:RESult][:MEASview<y>]:IRAMplitude? (on page 1318)	Returns the pulse impulse response amplitude measurement in the pulse table.
FETCh:PULSe[:RESult][:MEASview<y>]:IRTime? (on page 1319)	Returns the impulse response time in the pulse table.

Command	Description
FETCh:PULSe[:RESult][:MEASview<y>]:MFReqerror? (on page 1320)	Returns the maximum frequency error in the results table.
FETCh:PULSe[:RESult][:MEASview<y>]:MPHerror? (on page 1320)	Returns the maximum phase error in the results table.
FETCh:PULSe[:RESult][:MEASview<y>]:OVEDb? (on page 1321)	Returns the overshoot in dB in the pulse table.
FETCh:PULSe[:RESult][:MEASview<y>]:OVERshoot? (on page 1321)	Returns the overshoot in the pulse table.
FETCh:PULSe[:RESult][:MEASview<y>]:PHDeviation? (on page 1322)	Returns the phase deviation in the results table.
FETCh:PULSe[:RESult][:MEASview<y>]:PPFRequency? (on page 1323)	Returns the pulse-pulse carrier frequency in the results table.
FETCh:PULSe[:RESult][:MEASview<y>]:PPOWer? (on page 1324)	Returns the peak power in the results table.
FETCh:PULSe[:RESult][:MEASview<y>]:PPPHase? (on page 1325)	Returns the pulse-pulse carrier phase in the results table.
FETCh:PULSe[:RESult][:MEASview<y>]:RINTerval? (on page 1325)	Returns the repetition interval in the results table.
FETCh:PULSe[:RESult][:MEASview<y>]:RIPDb? (on page 1326)	Returns the ripple in dB in the pulse table.
FETCh:PULSe[:RESult][:MEASview<y>]:RIPPle? (on page 1327)	Returns the ripple in the pulse table.
FETCh:PULSe[:RESult][:MEASview<y>]:RISE? (on page 1327)	Returns the rise time in the results table.
FETCh:PULSe[:RESult][:MEASview<y>]:RMSFreqerror? (on page 1328)	Returns the RMS frequency error in the results table.
FETCh:PULSe[:RESult][:MEASview<y>]:RMSPherror? (on page 1328)	Returns the RMS phase error in the results table.
FETCh:PULSe[:RESult][:MEASview<y>]:RRATe? (on page 1329)	Returns the repetition rate in the results table.
FETCh:PULSe[:RESult][:MEASview<y>]:TIME? (on page 1329)	Returns the time in the results table.
FETCh:PULSe[:RESult][:MEASview<y>]:WIDTh? (on page 1330)	Returns the pulse width in the results table.
FETCh:PULSe:STATistics[:MEASview<y>]:ATX? (on page 1332)	Returns the average transmitted power of the statistics.

Command	Description
FETCh:PULSe:STATistics[:MEASview<y>]:AVERage? (on page 1333)	Returns the average on power of the statistics.
FETCh:PULSe:STATistics[:MEASview<y>]:DRODb? (on page 1333)	Returns the droop in dB of the statistics.
FETCh:PULSe:STATistics[:MEASview<y>]:DROop? (on page 1334)	Returns the droop of the statistics
FETCh:PULSe:STATistics[:MEASview<y>]:DUTPct? (on page 1335)	Returns the duty factor (%) of the statistics.
FETCh:PULSe:STATistics[:MEASview<y>]:DUTRatio? (on page 1335)	Returns the duty factor (ratio) of the statistics.
FETCh:PULSe:STATistics[:MEASview<y>]:FALL? (on page 1336)	Returns the fall time of the statistics.
FETCh:PULSe:STATistics:[:MEASview<y>]FDELta? (on page 1337)	Returns the delta frequency measurement statistics.
FETCh:PULSe:STATistics[:MEASview<y>]:FRDeviation? (on page 1338)	Returns the frequency deviation of the statistics.
FETCh:PULSe:STATistics[:MEASview<y>]:IRAMplitude? (on page 1338)	Returns the Pulse Impulse Response amplitude measurement in the pulse statistics.
FETCh:PULSe:STATistics[:MEASview<y>]:IRTime? (on page 1339)	Returns the impulse response time in the pulse statistics.
FETCh:PULSe:STATistics[:MEASview<y>]:MFReqerror? (on page 1341)	Returns the maximum frequency error of the statistics.
FETCh:PULSe:STATistics[:MEASview<y>]:MPHerror? (on page 1341)	Returns the maximum phase error of the statistics.
FETCh:PULSe:STATistics[:MEASview<y>]:OVEDb? (on page 1342)	Returns the overshoot measurement result in dB in the pulse statistics.
FETCh:PULSe:STATistics[:MEASview<y>]:OVERshoot? (on page 1342)	Returns the overshoot measurement result in the pulse statistics.
FETCh:PULSe:STATistics[:MEASview<y>]:PHDeviation? (on page 1343)	Returns the phase deviation of the statistics.
FETCh:PULSe:STATistics[:MEASview<y>]:PPFRequency? (on page 1344)	Returns the pulse-pulse carrier frequency of the statistics.
FETCh:PULSe:STATistics[:MEASview<y>]:PPOWer? (on page 1345)	Returns the peak power of the statistics.
FETCh:PULSe:STATistics[:MEASview<y>]:PPPHase? (on page 1346)	Returns the pulse-pulse carrier phase of the statistics.

Command	Description
FETCh:PULSe:STATistics[:MEASview<y>]:RINTerval? (on page 1346)	Returns the repetition interval of the statistics.
FETCh:PULSe:STATistics[:MEASview<y>]:RIPDb? (on page 1347)	Returns the ripple in dB of the statistics.
FETCh:PULSe:STATistics[:MEASview<y>]:RIPPle? (on page 1348)	Returns the ripple of the statistics.
FETCh:PULSe:STATistics[:MEASview<y>]:RISE? (on page 1348)	Returns the rise time of the statistics.
FETCh:PULSe:STATistics[:MEASview<y>]:RMSFreqerror? (on page 1349)	Returns the RMS frequency error of the statistics.
FETCh:PULSe:STATistics[:MEASview<y>]:RMSPherror? (on page 1350)	Returns the RMS phase error of the statistics.
FETCh:PULSe:STATistics[:MEASview<y>]:RRATe? (on page 1350)	Returns the repetition rate of the statistics.
FETCh:PULSe:STATistics[:MEASview<y>]:WIDTh? (on page 1351)	Returns the pulse width in the pulse statistics.
FETCh:PULSe:STATistics[:MEASview<y>]:X? (on page 1351)	Returns the horizontal values of the trace data.
FETCh:PULSe:STATistics[:MEASview<y>]:XY? (on page 1352)	Returns the horizontal and vertical values of the trace data.
FETCh:PULSe:STATistics[:MEASview<y>][:Y]? (on page 1353)	Returns the vertical values of the trace data.
FETCh:PULSe:TRACe[:MEASview<y>]:X? (on page 1354)	Returns the time values of the pulse trace.
FETCh:PULSe:TRACe[:MEASview<y>]:XY? (on page 1354)	Returns the horizontal (time) and vertical value pairs of the pulse trace.
FETCh:PULSe:TRACe[:MEASview<y>][:Y]? (on page 1355)	Returns the vertical values of the pulse trace.
FETCh:PULSe:CUMulative:HISTogram[:MEASview<y>]:OUTLier: HIGHer? (on page 1303)	Returns the outliers value exceeding the upper limit.
FETCh:PULSe:CUMulative:HISTogram[:MEASview<y>]:OUTLier: LOWer? (on page 1304)	Returns the outliers value below the lower limit.
FETCh:PULSe:CUMulative:HISTogram[:MEASview<y>]:PULse: COUNt? (on page 1304)	Returns the number of pulses in the bin.
FETCh:PULSe:CUMulative:HISTogram[:MEASview<y>]:PULse:TOTal? (on page 1305)	Returns the total number of pulses.

Command	Description
FETCh:PULSe:CUMulative:STATistics[:MEASview<y>] (on page 1305)	Returns the PulseCount, Max, Max TimeStamp, Min, Min TimeStamp, Peak to Peak, Average and Standard Deviation values for the chosen measurement.
FETCh:PULSe:OGRAm[:MEASview<y>]:MAX:FRAMe:COUNt (on page 1308)	Returns the frame count value.
FETCh:PULSe:OGRAm[:MEASview<y>]:RBW (on page 1309)	Returns the resolution bandwidth value only when Frequency vs Time display is opened in the Pulse-Ogram display.
FETCh:PULSe[:RESult][:MEASview<y>]:ADVanced (on page 1309)	Returns the Pulse Count, Max, Max at Pulse number, Min, Min at Pulse number, Peak to Peak, Average and Standard Deviation values for the chosen measurement.
FETCh:PULSe[:RESult][:MEASview<y>]:ADVanced:RANGe (on page 1311)	Queries the Pulse count, Max, Max Time Stamp, Min, Min Time Stamp, Peak to Peak, Average, and Standard Deviation values for the chosen measurement in the Pulse Table display.
FETCh:PULSe[:RESult][:MEASview<y>]:FABS? (on page 1316)	Returns the computed absolute frequency from the carrier.
FETCh:PULSe[:RESult][:MEASview<y>]:PPFD? (on page 1322)	Returns the computed pulse to pulse frequency difference from the carrier.
FETCh:PULSe[:RESult][:MEASview<y>]:PPPD? (on page 1324)	Returns the computed pulse to pulse phase difference from the carrier.
FETCh:PULSe:STATistics[:MEASview<y>]:ADVanced (on page 1331)	Returns the Pulse Count, Max, Max at Pulse number, Min, Min at Pulse number, Peak to Peak, Average and Standard Deviation values for the chosen measurement.
FETCh:PULSe:STATistics[:MEASview<y>]:FABS (on page 1336)	Returns the absolute frequency.
FETCh:PULSe:STATistics[:MEASview<y>]:MAX:PULSe:NUMber (on page 1339)	Returns the maximum value at a certain pulse number.
FETCh:PULSe:STATistics[:MEASview<y>]:MIN:PULSe:NUMber? (on page 1340)	Returns the minimum value at a certain pulse number.
FETCh:PULSe:STATistics[:MEASview<y>]:PPFD (on page 1344)	Returns the pulse-pulse frequency difference.
FETCh:PULSe:STATistics[:MEASview<y>]:PPPD (on page 1345)	Returns the pulse-pulse phase difference.
FETCh:RFIN subgroup	**All measurements**
FETCh:RFIN:IQ? (on page 1356)	Returns time-domain IQ data for a specific acquisition data record.
FETCh:RFIN:IQ:HEADer? (on page 1357)	Returns the header information for a specific acquisition data record.
FETCh:RFIN:IQ:SCALe? (on page 1357)	Returns the internal RF linear data scaling factor contained in the .tiq file header

Command	Description
FETCh:RFIN:IQ:VHEader? (on page 1358)	Returns the verbose header information for a specific acquisition data record.
FETCh:RFIN:RECord:IDS? (on page 1359)	Returns the beginning and end ID numbers of acquisition data.
FETCh:SEM subgroup	**Spectral Emissions Mask measurements**
FETCh:SEM:MHITs:COUNt? (on page 1359)	Returns the number of rows in the results table
FETCh:SEM:MHITs<x>:AMPLitude:ABSolute? (on page 1360)	Returns the absolute amplitude of the specified mask hit signal.
FETCh:SEM:MHITs<x>:AMPLitude:RELative? (on page 1360)	Returns the relative amplitude of the specified mask hit signal.
FETCh:SEM:MHITs<x>:FREQuency? (on page 1361)	Returns the frequency of the specified mask hit signal.
FETCh:SEM:MHITs<x>:INTeg:ABSolute? (on page 1361)	Returns the absolute integration amplitude of the specified mask hit signal.
FETCh:SEM:MHITs<x>:INTeg:RELative? (on page 1362)	Returns the relative integration amplitude of the specified mask hit signal.
FETCh:SEM:MHITs<x>:MARGin:ABSolute? (on page 1362)	Returns the absolute margin for the specified mask hit signal.
FETCh:SEM:MHITs<x>:MARGin:RELative? (on page 1363)	Returns the relative margin for the specified mask hit signal.
FETCh:SEM:MHITs<x>:OFFSet? (on page 1363)	Returns the offset for the specified mask hit signal.
FETCh:SEM:PASS? (on page 1364)	Returns the pass/fail limit test result.
FETCh:SEM:REF:POWer? (on page 1364)	Returns the reference power level.
FETCh:SEM:SPECtrum:X? (on page 1365)	Returns the frequencies of the spectrum trace.
FETCh:SEM:SPECtrum:XY? (on page 1365)	Returns the frequency and amplitude pairs of the spectrum trace.
FETCh:SEM:SPECtrum[:Y]? (on page 1366)	Returns the amplitude of the spectrum trace.
FETCh:SGRam subgroup	**Spectrogram measurement**
FETCh:SGRam? (on page 1366)	Returns the spectrogram trace data.
FETCh:SIGNalvu subgroup	**Signal measurement**
FETCh:SIGNalvu:ACQuisition:COUPling (on page 1367)	Returns the coupling value of the acquisition.
FETCh:SIGNalvu:ACQuisition:MAX:RT:BW (on page 1367)	Returns the maximum real time bandwidth value of the instrument.

Command	Description
FETCH:SIGNalvu:ACQuisition[:SOURce<x>]:DOWNconverter:INJEction:HIGHSide:MAXFreq? (on page 1368)	Returns the maximum range of input center frequency in case of high injection side.
FETCH:SIGNalvu:ACQuisition[:SOURce<x>]:DOWNconverter:INJEction:HIGHSide:MINFreq? (on page 1368)	Returns the minimum range of input center frequency in case of high injection side.
FETCH:SIGNalvu:ACQuisition[:SOURce<x>]:DOWNconverter:INJEction:LOWSide:MAXFreq? (on page 1369)	Returns the maximum range of input center frequency in case of low injection side.
FETCH:SIGNalvu:ACQuisition[:SOURce<x>]:DOWNconverter:INJEction:LOWSide:MINFreq? (on page 1370)	Returns the minimum range of input center frequency in case of low injection side.
FETCh:SIGNalvu:ACQuisition:TERMination? (on page 1370)	Returns the termination value of the acquisition.
FETCh:SPECtrum subgroup	**Spectrum measurement**
FETCh:SPECtrum[:MEASview<y>]:TRACe:BITMap? (Query Only) (on page 1371)	Returns trace waveform data of the bitmap trace in the Spectrum measurement.
FETCh:SPECtrum:TRACe<x>? (on page 1371)	Returns the trace data in the Spectrum Analyzer measurement.
FETCh:SPURious subgroup	**Spurious measurement**
FETCh:SPURious:CARRier:POWer? (on page 1372)	Returns the carrier power.
FETCh:SPURious:COUNt? (on page 1372)	Returns the number of spurious signals.
FETCh:SPURious:PASS? (on page 1373)	Returns the pass/fail limit test result.
FETCh:SPURious:SPECtrum:X? (on page 1373)	Returns the frequencies of the spectrum trace.
FETCh:SPURious:SPECtrum:XY? (on page 1374)	Returns the frequency and amplitude pairs of the spectrum trace.
FETCh:SPURious:SPECtrum[:Y]? (on page 1374)	Returns the amplitudes of the spectrum trace.
FETCh:SPURious:SPUR<x>:AMPLitude:ABSolute? (on page 1375)	Returns the absolute amplitude of the specified spurious signal.
FETCh:SPURious:SPUR<x>:AMPLitude:RELative? (on page 1375)	Returns the relative amplitude of the specified spurious signal.
FETCh:SPURious:SPUR<x>:FREQuency:ABSolute? (on page 1376)	Returns the absolute frequency of the specified spurious signal.
FETCh:SPURious:SPUR<x>:FREQuency:RELative? (on page 1376)	Returns the relative frequency of the specified spurious signal.

Command	Description
FETCh:SPURious:SPUR<x>:LIMit:ABSolute? (on page 1377)	Returns the absolute amplitude of the limit for a spurious signal.
FETCh:SPURious:SPUR<x>:LIMit:RELative? (on page 1377)	Returns the relative amplitude of the limit for a spurious signal.
FETCh:SPURious:SPUR<x>:LIMit:VIOLation? (on page 1378)	Returns whether the specified spurious signal exceeds the limit or not.
FETCh:SPURious:SPUR<x>:RANGe? (on page 1378)	Returns the frequency range in which the spurious signal occurred.
FETCh:SQUality subgroup (Option 21 only)	**Signal quality measurement**
FETCh:SQUality[:MEASview<y>]:FREQuency:DEViation? (on page 1379)	Returns the frequency deviation in the signal quality measurement.
FETCh:SQUality[:MEASview<y>]:FREQuency:DEViation:TABLe? (on page 1379)	Returns the number of columns and the values in the frequency deviation table for a signal quality measurement.
FETCh:SQUality[:MEASview<y>]:FREQuency:ERRor? (on page 1380)	Returns the frequency error in the signal quality measurement.
FETCh:SQUality[:MEASview<y>]:GAIN:IMBalance? (on page 1381)	Returns the gain imbalance in the signal quality measurement.
FETCh:SQUality[:MEASview<y>]:ORIGin:OFFSet? (on page 1381)	Returns the origin offset in the signal quality measurement.
FETCh:SQUality[:MEASview<y>]:PEAK:EVM? (on page 1382)	Returns the peak EVM (%) in the signal quality measurement.
FETCh:SQUality[:MEASview<y>]:PEAK:EVM:DB? (on page 1382)	Returns the peak EVM (dB)) in the signal quality measurement.
FETCh:SQUality[:MEASview<y>]:PEAK:EVM:DB:OFFSet? (on page 1383)	Returns the peak offset EVM (dB) in the signal quality measurement.
FETCh:SQUality[:MEASview<y>]:PEAK:EVM:LOCation? (on page 1384)	Returns the time at which the EVM is peak.
FETCh:SQUality[:MEASview<y>]:PEAK:EVM:LOCation:OFFSet? (on page 1384)	Returns the time at which the offset EVM is peak.
FETCh:SQUality[:MEASview<y>]:PEAK:EVM:OFFSet? (on page 1385)	Returns the peak offset EVM (%) in the signal quality measurement.
FETCh:SQUality[:MEASview<y>]:PEAK:FERRor? (on page 1385)	Returns the peak FSK error in the signal quality measurement.
FETCh:SQUality[:MEASview<y>]:PEAK:MERRor? (on page 1387)	Returns the peak magnitude error (%) in the signal quality measurement.

Command	Description
FETCh:SQUality[:MEASview<y>]:PEAK:MERRor:DB? (on page 1386)	Returns the peak magnitude error (dB) in the signal quality measurement.
FETCh:SQUality[:MEASview<y>]:PEAK:MERRor:LOCation? (on page 1387)	Returns the time at which the magnitude error is peak.
FETCh:SQUality[:MEASview<y>]:PEAK:PERRor? (on page 1388)	Returns the peak phase error in the signal quality measurement.
FETCh:SQUality[:MEASview<y>]:PEAK:PERRor:LOCation? (on page 1388)	Returns the time at which the phase error is peak.
FETCh:SQUality[:MEASview<y>]:QUADrature:ERRor? (on page 1389)	Returns the quadrature error in the signal quality measurement.
FETCh:SQUality[:MEASview<y>]:RHO? (on page 1389)	Returns the ρ (waveform quality) value in the signal quality measurement.
FETCh:SQUality[:MEASview<y>]:RMS:EVM? (on page 1390)	Returns the RMS EVM (%) in the signal quality measurement.
FETCh:SQUality[:MEASview<y>]:RMS:EVM:DB? (on page 1390)	Returns the RMS EVM (dB) in the signal quality measurement.
FETCh:SQUality[:MEASview<y>]:RMS:EVM:DB:OFFSet? (on page 1391)	Returns the RMS offset EVM (dB) in the signal quality measurement.
FETCh:SQUality[:MEASview<y>]:RMS:EVM:OFFSet? (on page 1392)	Returns the RMS offset EVM (%) in the signal quality measurement.
FETCh:SQUality[:MEASview<y>]:RMS:FERRor? (on page 1392)	Returns the RMS FSK error percent in the signal quality measurement.
FETCh:SQUality[:MEASview<y>]:RMS:MER:DB? (on page 1393)	Returns the RMS MER (Modulation Error Ratio) in dB in the signal quality measurement.
FETCh:SQUality[:MEASview<y>]:RMS:MERRor? (on page 1393)	Returns the RMS magnitude error (%) in the signal quality measurement.
FETCh:SQUality[:MEASview<y>]:RMS:MERRor:DB? (on page 1394)	Returns the RMS magnitude error (dB) in the signal quality measurement.
FETCh:SQUality[:MEASview<y>]:RMS:PERRor? (on page 1394)	Returns the RMS phase error in the signal quality measurement.
FETCh:SQUality[:MEASview<y>]:SYMBol:LENGth? (on page 1395)	Returns the number of analyzed symbols.
FETCh:SQUality[:MEASview<y>]:SYMBol:RATE? (on page 1395)	Returns the value of the calculated symbol rate in Hz.
FETCh:SQUality[:MEASview<y>]:SYMBol:RATE:ERRor? (on page 1396)	Returns the value of the symbol rate error in percent (%).

Command	Description
FETCh:TDIagram subgroup (Option 21 only)	**Trellis diagram**
FETCh:TDIagram[:MEASview<y>]:FERRor? (on page 1397)	Returns the frequency error.
FETCh:TDIagram[:MEASview<y>]:TRACe? (on page 1397)	Returns the Trellis diagram trace data.
FETCh:TOVerview subgroup	**Time overview**
FETCh:TOVerview:MEASview<y>? (on page 1398)	Returns the trace data.
FETCh:WLAN subgroup	**Wireless LAN measurements**
FETCh:WLAN:CONSte:MAGNitude? (on page 1452)	Returns the magnitude data for the WLAN Constellation measurement.
FETCh:WLAN:CONSte:PHASe? (on page 1452)	Returns the phase values for the WLAN Constellation measurement.
FETCh:WLAN:CONSte:TYPE? (on page 1453)	Returns the type (context) for the WLAN Constellation measurement.
FETCh:WLAN:CONSte:VALue? (on page 1454)	Returns the value (decoded symbol) for the WLAN Constellation measurement.
FETCh:WLAN:CRESponse:MAGNitude? (on page 1454)	Returns the magnitude data of the WLAN Channel Response measurement.
FETCh:WLAN:CRESponse:PHASe? (on page 1455)	Returns the phase data of the WLAN Channel Response measurement.
FETCh:WLAN:EVM:TRACe<x>? (on page 1455)	Returns the trace data for the selected WLAN EVM trace.
FETCh:WLAN:FLATness:PASS? (on page 1456)	Verifies the average power level values of the subcarriers remain within the limits defined for a particular standard.
FETCh:WLAN:FLATness:RESult? (on page 1456)	Returns the summary results of each segment of the WLAN Spectral Flatness display.
FETCh:WLAN:FLATness:TRACe<x>? (on page 1457)	Returns the WLAN Spectral Flatness trace data.
FETCh:WLAN:MERRor:TRACe<x>? (on page 1457)	Returns the trace data for the selected WLAN Magnitude Error trace.
FETCh:WLAN:PERRor:TRACe<x>? (on page 1458)	Returns the trace data for the selected WLAN Phase Error trace.
FETCh:WLAN:PVTime:BURSt:POWer? (on page 1459)	Returns the average power of the burst packet.
FETCh:WLAN:PVTime:BURSt:WIDTh? (on page 1459)	Returns the value of the measured time width of the burst packet from Power-On to Power-Down.
FETCh:WLAN:PVTime:FALL:TIME? (on page 1460)	Returns the width of the falling edge.
FETCh:WLAN:PVTime:FALL:TIME:STARt? (on page 1460)	Returns the start time of the falling edge.

Command	Description
FETCh:WLAN:PVTime:FALL:TIME:STOP? (on page 1460)	Returns the stop time of the falling edge.
FETCh:WLAN:PVTime:FTIMe? (on page 1461)	Returns the width of the falling edge.
FETCh:WLAN:PVTime:RISE:TIME? (on page 1461)	Returns the width of the rising edge.
FETCh:WLAN:PVTime:RISE:TIME:STARt? (on page 1462)	Returns the start time of the rising edge.
FETCh:WLAN:PVTime:RISE:TIME:STOP? (on page 1462)	Returns the stop time of the rising edge.
FETCh:WLAN:PVTime:RTIMe? (on page 1462)	Returns the width of the rising edge.
FETCh:WLAN:PVTime:TRACe:X? (on page 1463)	Returns the horizontal values.
FETCh:WLAN:PVTime:TRACe:XY? (on page 1463)	Returns the horizontal value (time in seconds) and vertical value (power).
FETCh:WLAN:PVTime:TRACe[:Y]? (on page 1464)	Returns the vertical values (power).
FETCh:WLAN:STABle:VALUe? (on page 1464)	Returns the number of bytes in the symbol table.
FETCh:WLAN:SUMMary:BURSt:INDex? (on page 1465)	Returns the index of the analyzed packet.
FETCh:WLAN:SUMMary:BURSt:POWer? (on page 1465)	Returns the average power of all symbols in the packet.
FETCh:WLAN:SUMMary:BURSt:POWer:CFACtor? (on page 1466)	Returns the peak-to-average burst power factor.
FETCh:WLAN:SUMMary:CERRor? (on page 1466)	Returns the symbol clock error.
FETCh:WLAN:SUMMary:CPE? (on page 1467)	Returns the RMS magnitude error of the pilots.
FETCh:WLAN:SUMMary:EVM:PEAK:ALL? (on page 1467)	Returns the peak EVM value for all subcarriers or all chips.
FETCh:WLAN:SUMMary:EVM:PEAK:CHIP:ALL? (on page 1467)	Returns the chip at which the reported peak EVM value occurred.
FETCh:WLAN:SUMMary:EVM:PEAK:CHIP:ONEKchips? (on page 1468)	Returns the chip at which the reported peak EVM value over the first 1000 chips occurred.
FETCh:WLAN:SUMMary:EVM:PEAK:DATA? (on page 1468)	Returns the peak EVM value for the data subcarriers.
FETCh:WLAN:SUMMary:EVM:PEAK:ONEKchips? (on page 1469)	Returns the peak EVM value measured over the first 1000 chips.

Command	Description
FETCh:WLAN:SUMMary:EVM:PEAK:PILots? (on page 1469)	Returns the peak EVM value for the pilots in the WLAN measurement.
FETCh:WLAN:SUMMary:EVM:PEAK:SCARrier:ALL? (on page 1469)	Returns the subcarrier on which the ALL peak EVM value occurred.
FETCh:WLAN:SUMMary:EVM:PEAK:SCARrier:DATA? (on page 1470)	Returns the subcarrier on which the DATA peak EVM value occurred.
FETCh:WLAN:SUMMary:EVM:PEAK:SCARrier:PILots? (on page 1470)	Returns the subcarrier on which the PILOT peak EVM value occurred.
FETCh:WLAN:SUMMary:EVM:PEAK:SYMBol:ALL? (on page 1471)	Returns the symbol on which the reported EVM value occurred.
FETCh:WLAN:SUMMary:EVM:PEAK:SYMBol:DATA? (on page 1471)	Returns the symbol on which the DATA peak EVM value occurred.
FETCh:WLAN:SUMMary:EVM:PEAK:SYMBol:PILots? (on page 1471)	Returns the symbol on which the PILOT peak EVM value occurred.
FETCh:WLAN:SUMMary:EVM:RMS:ALL? (on page 1472)	Returns the RMS EVM data for all subcarriers.
FETCh:WLAN:SUMMary:EVM:RMS:AVERage? (on page 1472)	Returns the average RMS EVM value.
FETCh:WLAN:SUMMary:EVM:RMS:DATA? (on page 1473)	Returns the RMS EVM data for the data subcarriers.
FETCh:WLAN:SUMMary:EVM:RMS:MAXimum? (on page 1473)	Returns the maximum RMS EVM value.
FETCh:WLAN:SUMMary:EVM:RMS:ONEKchips? (on page 1473)	Returns the RMS EVM data for the 1k chips in the WLAN measurement measured over the first 1000 chips.
FETCh:WLAN:SUMMary:EVM:RMS:PILots? (on page 1474)	Returns the RMS EVM data for the pilot subcarriers.
FETCh:WLAN:SUMMary:FERRor? (on page 1474)	Returns the difference between the measured carrier frequency of the signal and the measurement frequency setting.
FETCh:WLAN:SUMMary:HEADer? (on page 1475)	Returns the received packet header information.
FETCh:WLAN:SUMMary:HTSig? (on page 1475)	Returns the decimal values of the received packet HT-SIGNAL
FETCh:WLAN:SUMMary:IQOFfset? (on page 1475)	Returns the average magnitude of the DC subcarrier level relative to total signal power.
FETCh:WLAN:SUMMary:LSIG? (on page 1476)	Returns the decimal values of the received packet (legacy) SIGNAL symbols fields.

Command	Description
FETCh:WLAN:SUMMary:PACKet? (on page 1476)	Returns the type and number of symbols in the Preamble and Data portions of the packet.
FETCh:WLAN:SUMMary:PACKet:DMODulation? (on page 1477)	Returns the modulation used in the Data symbols.
FETCh:WLAN:SUMMary:PACKet:FORMat? (on page 1477)	Returns the packet format setting.
FETCh:WLAN:SUMMary:PACKet:GUARd? (on page 1477)	Returns the packet guard interval used by the DATA symbols.
FETCh:WLAN:SUMMary:SIG? (on page 1478)	Returns the decimal values of the received packet SIGNAL symbols fields.
FETCh:WLAN:SUMMary:VHTSig:A? (on page 1478)	Returns the decimal values of the received packet VHT-SIGNAL(A) symbols fields.
FETCh:WLAN:SUMMary:VHTSig:B? (on page 1479)	Returns the decimal values of the received packet VHT-SIGNAL(B) symbols fields.
FETCh:LTE subgroup	
FETCh:LTE:ACLR:MHITs:COUNt? (on page 1177)	Returns the number of rows in the results table.
FETCh:LTE:ACLR:MHITs<x>:BANDwidth? (on page 1177)	Returns the bandwidth value of the channel (row).
FETCh:LTE:ACLR:MHITs<x>:CHANnel:NAME? (on page 1178)	Returns the channel name (row) of the specified index in the results table.
FETCh:LTE:ACLR:MHITs<x>:FREQuency:OFFSet? (on page 1178)	Returns the frequency offset of the specified channel (row).
FETCh:LTE:ACLR:MHITs<x>:INTeg:LOWEr:ABSolute? (on page 1179)	Returns the lower absolute value of the channel (row).
FETCh:LTE:ACLR:MHITs<x>:INTeg:LOWEr:RELative? (on page 1179)	Returns the lower relative value of the channel (row).
FETCh:LTE:ACLR:MHITs<x>:INTeg:UPPEr:ABSolute? (on page 1180)	Returns the upper absolute value of the channel (row).
FETCh:LTE:ACLR:MHITs<x>:INTeg:UPPEr:RELative? (on page 1180)	Returns the upper relative value of the channel (row).
FETCh:LTE:ACLR:REFerence:POWer? (on page 1180)	Returns the reference power level.
FETCh:LTE:ACLR:RESUlts:STATus? (on page 1181)	Returns the pass or fail status.
FETCh:LTE:ACLR:SPECtrum:X? (on page 1181)	Returns the frequencies of the spectrum trace.

Command	Description
FETCh:LTE:ACLR:SPECtrum:XY? (on page 1182)	Returns the frequency and amplitude pairs of the spectrum trace.
FETCh:LTE:ACLR:SPECtrum:Y? (on page 1182)	Returns the amplitude of the spectrum trace.
FETCh:LTE:CHSPectrum:OBW? (on page 1183)	Returns the Occupied Bandwidth (OBW) in Hz.
FETCh:LTE:CHSPectrum:POWer:CHANnel? (on page 1183)	Returns the Channel Power value.
FETCh:LTE:CHSPectrum:SPECtrum? (on page 1184)	Returns spectrum trace data.
FETCh:LTE:CONSte:CELL:ID? (on page 1184)	Queries the Cell ID value.
FETCh:LTE:CONSte:FREQuency:ERRor? (on page 1184)	Queries Frequency Error value in Hz.
FETCh:LTE:CONSte:GROUp:ID? (on page 1185)	Queries the Group ID value.
FETCh:LTE:CONSte:SECTor:ID? (on page 1185)	Queries the Sector ID value.
FETCh:LTE:PVTime:OFFSlot:POWer? (on page 1186)	Returns the TOff power measurement value in dBm/MHz.
FETCh:LTE:PVTime:RESUlts:STATus (on page 1186)	Returns the pass or fail status.
FETCh:LTE:PVTime:TRACe:X? (on page 1186)	Returns the horizontal values (time in seconds).
FETCh:LTE:PVTime:TRACe:XY? (on page 1187)	Returns the horizontal value (time in seconds) and vertical value (power).
FETCh:LTE:PVTime:TRACe:Y? (on page 1187)	Returns the vertical values (power).

Initiate commands

Use the INITiate commands to control the acquisition of data.

Initiate commands

Command	Description
INITiate:CONTinuous (on page 1480)	Sets or queries whether to acquire data continuously.
INITiate[:IMMediate] (on page 1480)	Starts input signal acquisition.
INITiate:RESume (on page 1481)	Starts data acquisition.

Input commands

Use the INPut commands to control the characteristics of the signal input.

Input commands

Command	Description
INPut:CORRection:EXTernal:EDIT<x>:INTerpolation (on page 1482)	Sets or queries the interpolation setting to use with the indicated external gain table.
INPut:CORRection:EXTernal:EDIT<x>:NEW (on page 1483)	Creates the indicated external loss table.
INPut:CORRection:EXTernal:TYPE (on page 1485)	Sets or queries the data type to use when applying the external loss table corrections.
INPut:{MLEVel|RLEVel} (on page 1486)	Sets or queries the reference level.
INPut[:RF]:ATTenuation (on page 1486)	Sets or queries the input attenuation.
NPut[:RF]:ATTenuation:AUTO (on page 1486)	Sets or queries whether to set the attenuation automatically.
INPut[:RF]:ATTenuation:MONitor:STATe (on page 1487)	Sets or queries whether to enable to monitor attenuator use.
INPut[:RF]:GAIN:STATe (on page 1487) (Option 50 or 51 only)	Sets or queries whether to enable the internal preamplifier.
INPut:TYPE (on page 1488)	Sets or queries the source type.

Input commands

Mass memory commands

Use the MMEMory commands to manipulate files on the mass memory devices.See the Index for commands not listed in the following table.

For the trace specifier TRACe<x>, refer to Trace Mnemonics (on page 336).

Not all of these commands apply to SignalVu. See the individual command descriptions for details.

Mass memory (MMEMory) commands

Command	Description
MMEMory:{AM|FM|PM} subgroup	**AM/FM/PM measurement**
MMEMory:{AM|FM|PM}:LOAD:TRACe (on page 1488)	Loads trace data from the specified file.
MMEMory:TXGain subgroup	**Transmission Gain measurements**
MMEMory:TXGain:LOAD:TRACe<x> (on page 1539)	Loads the specified saved trace from a file.
MMEMory:TXGain:SHOW:TRACe<x> (on page 1539)	Displays a saved trace instead of a live trace.
MMEMory:TXGain:STORe:TRACe<x> (on page 1540)	Saves the selected trace to a file for later recall and analysis.

Command	Description
MMEMory:Noise subgroup	**Noise Figure and Gain measurements**
MMEMory:NOISe:GAIN:LOAD:TRACe (on page 1517)	Loads the Noise Gain trace data from the specified file.
MMEMory:NOISe:GAIN:SHOW:TRACe (on page 1518)	Show or hide the Noise Gain trace from the specified file.
MMEMory:NOISe:GAIN:STORe:TRACe (on page 1518)	Stores the Noise Gain trace data in the specified file.
MMEMory:Noise:LOAD:ENRCALibration (on page 1519)	Loads the specified Noise Figure ENR table.
MMEMory:Noise:LOAD:FREQTABle (on page 1519)	Loads the specified Noise Figure Frequency table.
MMEMory:Noise:STORe:ENRCALibration (on page 1520)	Saves the Noise Figure ENR table to the specified location.
MMEMory:Noise:STORe:FREQTABle (on page 1520)	Saves the Noise Figure Frequency table to the specified location.
MMEMory:{AM|FM|PM}:SHOW:TRACe<x> (on page 1489)	Enables display of a recalled trace file.
MMEMory:{AM|FM|PM}:STORe:TRACe (on page 1489)	Stores trace data in the specified file.
MMEMory:AUDio subgroup	**Audio measurements**
MMEMory:AUDio:LOAD:FILTer (on page 1489)	Loads a user-defined audio filter from the specified file.
MMEMory:AVTime subgroup	**Amplitude versus Time measurement**
MMEMory:AVTime[:MEASview<y>]:LOAD:TRACe<x> (on page 1490)	Loads trace data from the specified file.
MMEMory:AVTime[:MEASview<y>]:SHOW:TRACe<x> (on page 1490)	Enables display of a recalled trace file.
MMEMory:AVTime[:MEASview<y>]:STORe:TRACe<x> (on page 1492)	Stores trace data in the specified file.
MMEMory:CALibration subgroup	**Measurement calibration**
MMEMory:CALibration:LOAD:CORRection:EXTernal:EDIT<x> (on page 1492)	Loads the external loss table data from a specified file.
MMEMory:CALibration:STORe:CORRection:EXTernal:EDIT<x> (on page 1493)	Stores an external loss table to a specified file.
MMEMory:CCDF subgroup	**CCDF measurement**
MMEMory:CCDF:LOAD:TRACe<x> (on page 1493)	Loads trace data from the specified file.

Command	Description
MMEMory:CCDF:SHOW:TRACe<x> (on page 1494)	Enables display of a recalled trace file.
MMEMory:CCDF:STORe:TRACe<x> (on page 1494)	Stores trace data in the specified file.
MMEMory:DDEMod subgroup	**DPX spectrum measurement**
MMEMory:DDEMod[:SOURce<x>]:LOAD:FILTer:MEASurement:USER<y> (on page 1495)	Loads a user-defined measurement filter from the specified file.
MMEMory:DDEMod[:SOURce<x>]:LOAD:FILTer:MEASurement:UOTHer (on page 1495)	Loads a user-defined measurement filter from the specified file.
MMEMory:DDEMod[:SOURce<x>]:LOAD:FILTer:REFerence:USER<y> (on page 1495)	Loads the specified user-defined reference filter.
MMEMory:DDEMod[:SOURce<x>]:LOAD:FILTer:REFerence:UOTHer (on page 1496)	Loads the specified user-defined reference filter
MMEMory:DDEMod[:SOURce<x>]:LOAD:SYMBol:MAP (on page 1497)	Loads the specified symbol map filename.
MMEMory:DPX subgroup	**DPX spectrum measurement**
MMEMory:DPX:TDM:FREQuency:TXBer:LOAD:PATTern (on page 1498)	Loads the Tx BER User data pattern from the specified file.
MMEMory:DPX:LOAD:TRACe<x> (on page 1497)	Loads trace data from the specified file.
MMEMory:DPX:SHOW:TRACe<x> (on page 1497)	Enables display of a recalled trace file.
MMEMory:DPX:STORe:TRACe<x> (on page 1498)	Stores trace data in the specified file.
MMEMory:{FSETtling|PSETtling} subgroup	**Frequency and Phase Settling time measurements**
[MMEMory:{FSETtling|PSETtling}:LOAD:TRACe<x> (on page 1507)	Loads the Frequency or Phase Settling Time trace data from the specified file.
[MMEMory:{FSETtling|PSETtling}:SHOW:TRACe<x> (on page 1507)	Enables display of a recalled Frequency or Phase Settling trace file in Trace<x>.
[MMEMory:{FSETtling|PSETtling}:STORe:TRACe<x> (on page 1508)	Stores the Frequency or Phase Settling trace data in the specified file.
MMEMory:FVTime subgroup	**Frequency versus Time measurement**
MMEMory:FVTime[:MEASview<y>]:LOAD:TRACe (on page 1508)	Loads trace data from the specified file.
MMEMory:FVTime[:MEASview<y>[:SHOW:TRACe<x> (on page 1509)	Enables display of a recalled trace file.
MMEMory:FVTime[:MEASview<y>]:STORe:TRACe (on page 1509)	Stores trace data in the specified file.

Command	Description
MMEMory:IQVTime subgroup	**RF I&Q versus Time measurement**
MMEMory:IQVTime[:MEASview<y>]:LOAD:TRACe:I (on page 1510)	Loads I trace data from the specified file.
MMEMory:IQVTime[:MEASview<y>]:LOAD:TRACe:Q (on page 1510)	Loads Q trace data from the specified file.
MMEMory:IQVTime[:MEASview<y>]:SHOW:TRACe:I (on page 1511)	Loads trace data from the specified file.
MMEMory:IQVTime[:MEASview<y>]:SHOW:TRACe<x>:Q (on page 1511)	Enables display of a recalled trace file.
MMEMory:IQVTime[:MEASview<y>]:STORe:TRACe:I (on page 1512)	Stores I trace data in the specified file.
MMEMory:IQVTime[:MEASview<y>]:STORe:TRACe:Q (on page 1512)	Stores Q trace data in the specified file.
MMEMory:LTE subgroup	
MMEMory:LTE:ACLR:LOAD:TABLe (on page 1515)	Loads the LTE ACLR table containing the limits for enabled ranges from the specified .cvs file.
MMEMory:LTE:ACLR:STORe:TABLe (on page 1516)	Stores the LTE ACLR table containing the limits for enabled ranges in a specified .cvs file, allowing you to export the file into Microsoft Excel or other database system.
MMEMory:LOAD subgroup	**Load commands**
MMEMory:LOAD:IQ (on page 1513)	Loads time-domain IQ waveform from a specified file.
MMEMory:LOAD:STATe (on page 1513)	Loads the instrument setup from a specified file.
MMEMory:LOAD:TRACe (on page 1513)	Loads the trace data from a specified file.
MMEMory:OFDM CONSTe subgroup	**OFDM Constellation commands**
MMEMory:OFDM:CONSte:LOAD:TRACe (on page 1520)	Loads trace data from the specified file.
MMEMory:OFDM:CONSTe:SHOW:TRACe (on page 1521)	Enables the display of a recalled trace file.
MMEMory:OFDM:CONSte:STORe:TRACe (on page 1521)	Stores trace data in the specified file.
MMEMory:PHVTime subgroup	**Phase versus Time measurement**
MMEMory:PHVTime[:MEASview<y>]:LOAD:TRACe (on page 1522)	Loads trace data from the specified file.

Command	Description
MMEMory:PHVTime[:MEASview<y>]:SHOW:TRACe (on page 1522)	Enables display of a recalled trace file.
MMEMory:PHVTime[:MEASview<y>]:STORe:TRACe (on page 1523)	Stores trace data in the specified file.
MMEMory:PNOise subgroup	**Phase noise measurement**
MMEMory:PNOise:LOAD:TRACe<x> (on page 1523)	Loads the Phase Noise trace data from the specified file.
MMEMory:PNOise:SHOW:TRACe<x> (on page 1523)	Enables display of a recalled Phase Noise trace file.
MMEMory:PNOise:STORe:TRACe<x> (on page 1524)	Stores the Phase Noise trace data in the specified file.
MMEMory:SEM subgroup	**Spectral Emissions Mask measurement**
MMEMory:SEM:LOAD:TABLe (on page 1524)	Loads the SEM table from the specified file.
MMEMory:SEM:STORe:TABLe (on page 1525)	Stores the SEM table in the specified file.
MMEMory:SGRam subgroup	**Spurious measurement**
MMEMory:SGRam[:MEASview<y>]:LOAD:TRACe (on page 1525)	Loads the trace data from a specified file.
MMEMory:SGRam[:MEASview<y>]:SHOW:TRACe (on page 1526)	Enables display of a recalled trace file.
MMEMory:SGRam[:MEASview<y>]:STORe:TRACe (on page 1525)	Stores the Spectrogram trace data in the specified file.
MMEMory:SPECtrum subgroup	**Spectrum measurement**
MMEMory:SPECtrum[:MEASview<y>]:LOAD:TRACe (on page 1528)	Loads the trace data from a specified file.
MMEMory:SPECtrum[:MEASview<y>]:SHOW:TRACe<x> (on page 1528)	Enables display of a recalled trace file.
MMEMory:SPECtrum[:MEASview<y>]:STORe:TRACe<x>	Stores the spectrum trace data in the specified file.
MMEMory:SPURious subgroup	**Spurious measurement**
MMEMory:SPURious:LOAD:TABLe (on page 1530)	Loads the spurious table from the specified file.
MMEMory:SPURious:STORe:TABLe (on page 1532)	Stores the spurious table in the specified file.
MMEMory:STORe subgroup	**Store commands**

Command	Description
MMEMory:STORe:IQ (on page 1533)	Saves time-domain IQ waveform in the acquisition memory to a specified file.
MMEMory:STORe:IQ:CSV (on page 1533)	Saves time-domain IQ waveform in the acquisition memory to a specified file in the CSV (Comma Separated Values) format.
MMEMory:STORe:IQ:MAT (on page 1533)	Saves time-domain IQ waveform in the acquisition memory to a specified file in the MATLAB format.
MMEMory:STORe:IQ:SELEct:DATA (on page 1534)	Sets or queries the type of data stored in an acquisition data file.
MMEMory:STORe:IQ:SELEct:FRAMes (on page 1534)	Sets or queries which frames to save when saving TIQ acquisition data files.
MMEMory:STORe:IQ:SELEct:LENGth (on page 1535)	Sets or queries what portion of each record is saved when saving acquisition data.
MMEMory:STORe:MSTate (on page 1535)	Stores the measurement parameters in a specified file.
MMEMory:STORe:RESults (on page 1535)	Stores the measurement results including measurement parameters and trace data to a specified file.
MMEMory:STORe:SCReen (on page 1536)	Stores the measurement results in a specified file.
MMEMory:STORe:STATe (on page 1536)	Stores the instrument setup in a specified file.
MMEMory:STORe:TRACe (on page 1537)	Stores trace data in a specified file.
MMEMory:TOVerview subgroup	**Spurious measurement**
MMEMory:TOVerview[:MEASview<y>]:LOAD:TRACe1 (on page 1537)	Loads the trace data from a specified file.
MMEMory:TOVerview[:MEASview<y>]:SHOW:TRACe1 (on page 1538)	Enables display of a recalled trace file.
MMEMory:TOVerview[:MEASview<y>]:STORe:TRACe1 (on page 1538)	Stores the selected trace data into the specified file.
MMEMory:WLAN subgroup	**Wireless LAN measurements**
MMEMory:WLAN:CRESponse:LOAD:TRACe<x> (on page 1542)	Loads the selected WLAN Channel Response trace data from the specified file.
MMEMory:WLAN:CRESponse:SHOW:TRACe<x> (on page 1542)	Enables the display of a recalled WLAN Channel Response trace.
MMEMory:WLAN:CRESponse:STORe:TRACe<x> (on page 1543)	Stores the WLAN Channel Response measurement trace data in the specified file.

Mass memory (MMEMory) commands

Specifying the File

For loading and storing a file, specify the file following these rules

- You can omit the file extension to load and store data. The measurement-specific extension is automatically added.
- You can use the absolute path to specify the file name. For example,specify the SAMPLE1 file in the My Documents folder on the C drive as `"C:\My Documents\SAMPLE1"`.

initially.

Once a file is saved to a different directory, the new directory will be used as the default for all load and store operations.

Output commands

Use the OUTPut commands to control the characteristics of the signal output. See the Index for commands not listed in the following table.

Output commands

Command		Description
OUTPut:IF:{BANDwidth|BWIDth} (on page 1546)	(Option 05 only)	Sets or queries the IF output filter.
OUTPut:IF[:STATe] (on page 1546)	(Option 05 only)	Sets or queries whether to turn on or off IF output.
OUTPut:IQ[:STATe] (on page 1547)	(Option 05 only)	Sets or queries whether to turn on or off IQ output.
OUTPut:IQ:SOURce (on page 1547)	(Option 05 only)	Sets or queries the IQ data transfer via the LVDS card.
OUTPut:NOISe[:STATe] (on page 1547)		Sets or queries whether to turn on or off 28 V DC power.
OUTPut:ZS:{BANDwidth|BWIDth} (on page 1550)		Sets or queries the resolution bandwidth RBW filter.
OUTPut:ZS:{BANDwidth|BWIDth}:AUTO (on page 1550)		Sets or queries whether the resolution bandwidth will be set manually or automatically.
OUTPut:ZS:{BANDwidth|BWIDth}:STATe (on page 1551)		Sets or queries the state of the resolution bandwidth filter.
OUTPut:ZS[:STATe] (on page 1552)		Enable or disable the Zero Span output.
OUTPut:ZS:DELay? (on page 1551)		Queries the delay from the input connector to the Zero Span output.
OUTPut:ZS:FILTer:STATe (on page 1552)		Sets or queries the signal path of the filter to the Zero Span output.
OUTPut:ZS:VOLTage (on page 1552)		Sets or queries the voltage level at the Zero Span output.

Command		Description
OUTPut:ZS:VOLTage:AUTO (on page 1553)		Sets or queries or queries how the output voltage is referenced.
OUTPut:TGen:BRIDge:MODE (on page 1548)	(Option 04 only; available for the RSA500A/600A series only))	Sets or queries the tracking generator bridge mode.
OUTPut:TGen:ENABle (on page 1548)	(Option 04 only; available for the RSA500A/600A series only))	Sets or queries the power state of the tracking generator.
OUTPut:TGen:FREQuency (on page 1549)	(Option 04 only; available for the RSA500A/600A series only))	Sets or queries the tracking generator frequency in Hz.
OUTPut:TGen:INSTalled? (on page 1549)	(Option 04 only; available for the RSA500A/600A series only))	Queries for the presence of the tracking generator hardware.
OUTPut:TGen:LEVel (on page 1550)	(Option 04 only; available for the RSA500A/600A series only))	Sets or queries the tracking generator output level in dBm.
OUTPut:IF:{BANDwidth|BWIDth} (on page 1546)	(Option 05 only)	Sets or queries the IF output filter.
OUTPut:IF[:STATe] (on page 1546)	(Option 05 only)	Sets or queries whether to turn on or off IF output.
OUTPut:IQ[:STATe] (on page 1547)	(Option 05 only)	Sets or queries whether to turn on or off IQ output.
OUTPut:NOISe[:STATe] (on page 1547)		Sets or queries whether to turn on or off 28 V DC power.
OUTPut:ZS:{BANDwidth|BWIDth} (on page 1550)		Sets or queries the resolution bandwidth RBW filter.
OUTPut:ZS:{BANDwidth|BWIDth}:AUTO (on page 1550)		Sets or queries whether the resolution bandwidth will be set manually or automatically.
OUTPut:ZS:{BANDwidth|BWIDth}:STATe (on page 1551)		Sets or queries the state of the resolution bandwidth filter.
OUTPut:ZS[:STATe] (on page 1552)		Enable or disable the Zero Span output.
OUTPut:ZS:DELay? (on page 1551)		Queries the delay from the input connector to the Zero Span output.
OUTPut:ZS:FILTer:STATe (on page 1552)		Sets or queries the signal path of the filter to the Zero Span output.
OUTPut:ZS:VOLTage (on page 1552)		Sets or queries the voltage level at the Zero Span output.
OUTPut:ZS:VOLTage:AUTO (on page 1553)		Sets or queries or queries how the output voltage is referenced.

Command		Description
OUTPut:TGen:BRIDge:MODE (on page 1548)	(Option 04 only; available for the RSA500A/600A series only))	Sets or queries the tracking generator bridge mode.
OUTPut:TGen:ENABle (on page 1548)	(Option 04 only; available for the RSA500A/600A series only))	Sets or queries the power state of the tracking generator.
OUTPut:TGen:FREQuency (on page 1549)	(Option 04 only; available for the RSA500A/600A series only))	Sets or queries the tracking generator frequency in Hz.
OUTPut:TGen:INSTalled? (on page 1549)	(Option 04 only; available for the RSA500A/600A series only))	Queries for the presence of the tracking generator hardware.
OUTPut:TGen:LEVel (on page 1550)	(Option 04 only; available for the RSA500A/600A series only))	Sets or queries the tracking generator output level in dBm.

Read commands

The READ commands acquire an input signal once in the single mode and obtain the measurement results from that data. See the Index for commands not listed in the following table.

To fetch the measurement results from the data currently residing in the memory without acquiring the input signal, use the FETCh commands.

Read commands

Command	Description
READ:ACPower subgroup	**Channel power and ACPR measurement**
READ:ACPower? (on page 1553)	Returns the ACPR measurement results.
READ:PULSe subgroup	**Pulse measurements**
READ:PULSe:CUMulative:HISTogram[:MEASview<y>]:OUTLier:HIGHer? (on page 1763)	Returns the outliers value exceeding the upper limit.
READ:PULSe:CUMulative:HISTogram[:MEASview<y>]:OUTLier:LOWer? (on page 1764)	Returns the outliers value below the lower limit.
READ:PULSe:CUMulative:HISTogram[:MEASview<y>]:PULse:COUNt? (on page 1764)	Returns the number of pulses in the bin.
READ:PULSe:CUMulative:HISTogram[:MEASview<y>]:PULse:TOTal? (on page 1765)	Returns the total number of pulses.
READ:PULSe:CUMulative:STATistics[:MEASview<y>] (on page 1765)	Returns the PulseCount, Max, Max TimeStamp, Min, Min TimeStamp, Peak to Peak, Average and Standard Deviation values for the chosen measurement.
READ:PULSe[:RESult][:MEASview<y>]:ADVanced (on page 1768)	Returns the Pulse Count, Max, Max at Pulse number, Min, Min at Pulse number, Peak to Peak, Average and Standard Deviation values for the chosen measurement in pulse table.

Command	Description
READ:PULSe[:RESult][:MEASview<y>]:ADVanced:RANGe (on page 1770)	Queries the Pulse Count, Max, Max TimeStamp, Min, Min TimeStamp, Peak to Peak, Average, and Standard Deviation values for the chosen measurement in the Pulse Table display.
READ:PULSe[:RESult][:MEASview<y>]:FABS? (on page 1776)	Returns the computed absolute frequency from the carrier in the pulse table.
READ:PULSe[:RESult][:MEASview<y>]:PPFD? (on page 1783)	Returns the computed pulse to pulse frequency difference from the carrier in the pulse table.
READ:PULSe[:RESult][:MEASview<y>]:PPPD? (on page 1784)	Returns the computed pulse to pulse phase difference from the carrier in the pulse table.
READ:PULSe:STATistics[:MEASview<y>]:ADVanced (on page 1792)	Returns the Pulse Count, Max, Max at Pulse number, Min, Min at Pulse number, Peak to Peak, Average and Standard Deviation values for the chosen measurement in pulse statistics.
READ:PULSe:STATistics[:MEASview<y>]:FABS (on page 1798)	Returns the absolute frequency in pulse statistics.
READ:PULSe:STATistics[:MEASview<y>]:MAX:PULSe:NUMber (on page 1802)	Returns the maximum value at a certain pulse number in pulse statistics.
READ:PULSe:STATistics[:MEASview<y>]:MIN:PULSe:NUMber? (on page 1803)	Returns the minimum value at a certain pulse number in pulse statistics.
READ:PULSe:STATistics[:MEASview<y>]:PPFD (on page 1806)	Returns the pulse-pulse frequency difference in pulse statistics.
READ:PULSe:STATistics[:MEASview<y>]:PPPD (on page 1808)	Returns the pulse-pulse phase difference in pulse statistics.
READ:Bluetooth subgroup	**Bluetooth measurements**
READ:BIBEmissions:FTX? (on page 1579)	Returns the band number corresponding to the transmitted FTX detected in the display.
READ:BIBEmissions:POWer? (on page 1579)	Returns all channel power results.
READ:BIBEmissions:RESUlts:STATus? (on page 1580)	Returns the Pass/Fail result status.
READ:BLUEtooth:CONSte:FERRor? (on page 1580)	Queries the frequency error.
READ:BLUEtooth:CONSte:TRACe? (on page 1581)	Returns the trace data for the display.
READ:BLUEtooth:EDIagram:FDEViation? (on page 1581)	Returns the frequency deviation vs. time trace data with the X values.
READ:BLUEtooth:EDIagram:FERRor? (on page 1582)	Returns the frequency error.

Command	Description
READ:BLUEtooth:EDIagram:HORiz:SCALe? (on page 1582)	Returns the value of the horizontal scale.
READ:BLUEtooth:EDIagram:I? (on page 1583)	Returns the I vs. time trace.
READ:BLUEtooth:EDIagram:Q? (on page 1583)	Returns the Q vs. time trace.
READ:BLUEtooth:FDVTime:TRACe? (on page 1584)	Returns the trace data or the display.
READ:BLUEtooth:FREQuency:ERROr? (on page 1584)	Returns the frequency error.
READ:BLUEtooth:FREQuency:ERROr:TYPE (on page 1585)	Returns the frequency error type.
READ:BLUEtooth:FREQuency:OFFSet:DRIFt:F1FZero? (on page 1585)	Returns the drift in frequency offset (f1–f0 — the frequency offset calculated in the first interval in the payload — the frequency offset calculated in the preamble).
READ:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFN5? (on page 1586)	Returns the maximum drift of the frequency offset in payload intervals spaced 50 µs away.
READ:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFN5:INTErval? (on page 1586)	Returns the interval at which the maximum drift of frequency offset in a duration of 50 µs occurred.
READ:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFZERO? (on page 1587)	Returns the maximum drift of the frequency offset of the intervals in the payload from the preamble packet in the Bluetooth Frequency Deviation vs. Time display.
READ:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFZERO:INTErval? (on page 1587)	Returns the interval (n) at which the maximum drift fn-f0 occurred in the Bluetooth Frequency Deviation vs. Time display.
READ:BLUEtooth:FREQuency:OFFSet:MAX? (on page 1588)	Returns the maximum frequency offset in the Frequency Deviation vs. Time and the Center Frequency Offset and Drift displays.
READ:BLUEtooth:FREQuency:OFFSet:MAX:INTErval? (on page 1588)	Returns the maximum frequency offset interval in the Frequency Deviation vs. Time and the Center Frequency Offset and Drift displays.
READ:BLUEtooth:FREQuency:OFFSet:PREAmble? (on page 1589)	Returns the offset calculated in the preamble region in the Frequency Deviation vs. Time and the Center Frequency Offset and Drift displays.
READ:BLUEtooth:MODUlation:CHARacteristics:AVErage:F? (on page 1589)	Returns the average frequency value when the selected test pattern is Other in the Bluetooth Frequency Deviation vs. Time display.
READ:BLUEtooth:MODUlation:CHARacteristics:AVErage:F2F1ratio? (on page 1590)	Returns the average ratio value for the high and low frequency deviation in the Bluetooth Frequency Deviation vs. Time display.
READ:BLUEtooth:MODUlation:CHARacteristics:AVErage:FONE? (on page 1590)	Returns the average frequency deviation for the low deviation pattern in the Bluetooth Frequency Deviation vs. Time display.
READ:BLUEtooth:MODUlation:CHARacteristics:AVErage:FTWO? (on page 1591)	Returns the average frequency deviation for the high deviation pattern.

Command	Description
READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:COUNt? (on page 1591)	Returns the number of packets analyzed to obtain average drift results.
READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:F1FZero? (on page 1592)	Returns the drift f1–f0 value.
READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:F1FZero:STATus? (on page 1592)	Returns the Pass/Fail drift f1–f0 result status in the packet.
READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFN5? (on page 1593)	Returns the maximum drift of the fn–fn-5 value.
READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFN5:STATus? (on page 1593)	Returns the Pass/Fail status of the maximum fn–fn-5 drift in the packet.
READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFZero? (on page 1594)	Returns the maximum drift fn–f0.
READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFZero:STATus? (on page 1594)	Returns the Pass/Fail result status of the fn–f0 maximum drift in the packet.
READ:BLUEtooth:SUMMary:FREQuency:OFFSet:MAX? (on page 1598)	Returns the maximum frequency offset present in the packet.
READ:BLUEtooth:SUMMary:FREQuency:OFFSet:MAX:STATus? (on page 1599)	Returns the Pass/Fail result status of the frequency offset maximum in the packet.
READ:BLUEtooth:SUMMary:FREQuency:OFFSet:PREAmble? (on page 1599)	Returns the frequency offset value in the preamble region.
READ:BLUEtooth:SUMMary:FREQuency:OFFSet:PREAmble:STATus? (on page 1600)	Returns the Pass/Fail result status of the frequency offset in the preamble region.
READ:BLUEtooth:SUMMary:MODULation:CHARacteristics:AVERage:DELTa:F:MAX:PERCentage:COUNt? (on page 1600)	Returns the number of packets analyzed out of 10 packets for ΔFavg.
READ:BLUEtooth:SUMMary:MODULation:CHARacteristics:AVERage:DELTa:FONE:MAX:PERCentage:COUNt? (on page 1601)	Returns the number of packets analyzed out of 10 packets for the ΔF1avg.
READ:BLUEtooth:SUMMary:MODULation:CHARacteristics:AVERage:DELTa:FTWO:MAX:PERCentage? (on page 1601)	Returns the percentage of the ΔF2max% values that are above the specified limit.
READ:BLUEtooth:SUMMary:MODULation:CHARacteristics:AVERage:DELTa:FTWO:MAX:PERCentage:COUNt? (on page 1602)	Returns the number of packets analyzed out of 10 packets for ΔF2avg.
READ:BLUEtooth:SUMMary:MODULation:CHARacteristics:AVERage:DELTa:FTWO:MAX:PERCentage:STATus? (on page 1602)	Returns the Pass/Fail result status of the ΔF2Max% field.

Command	Description
READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:F? (on page 1604)	Returns the ΔFavg value.
READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:F2F1ratio? (on page 1603)	Returns the ratio of the ΔF2avg to ΔF1avg.
READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:F2F1ratio:STATus? (on page 1603)	Returns the Pass/Fail results status of the ΔF2avg/ΔF1avg field.
READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:FONE? (on page 1604)	Returns the delta F1 average value.
READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:FONE:STATus? (on page 1605)	Returns the Pass/Fail results status of the delta F1 average.
READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:FTWO? (on page 1605)	Returns the delta F2 average value,
READ:BLUEtooth:SUMMary:OUTPut:POWEr:AVErage? (on page 1606)	Returns the average output power.
READ:BLUEtooth:SUMMary:OUTPut:POWEr:AVErage:STATus? (on page 1607)	Returns the Pass/Fail results status of the average output power.
READ:BLUEtooth:SUMMary:OUTPut:POWEr:COUNt? (on page 1607)	Returns the number of packets analyzed out of 10 packets for the calculation of the output power.
READ:BLUEtooth:SUMMary:OUTPut:POWEr:EDR:COUNt? (on page 1608)	Returns the number of packets analyzed out of 10 packets for calculation of the output power for EDR.
READ:BLUEtooth:SUMMary:OUTPut:POWEr:EDR:DPSK? (on page 1608)	Returns the power in the DPSK portion of the EDR burst.
READ:BLUEtooth:SUMMary:OUTPut:POWEr:EDR:GPSK? (on page 1609)	Returns the power in the GFSK portion of the EDR burst.
READ:BLUEtooth:SUMMary:OUTPut:POWEr:PEAK? (on page 1609)	Returns the peak output power.
READ:BLUEtooth:SUMMary:OUTPut:POWEr:PEAK:STATus? (on page 1610)	Returns the Pass/Fail results status of the peak output power.
READ:BLUEtooth:SUMMary:PACKet:BDR:CRC? (on page 1610)	Returns the Basic Rate value of the CRC field.
READ:BLUEtooth:SUMMary:PACKet:BDR:HEADer:ARQN? (on page 1611)	Returns the value of ARQN field.
READ:BLUEtooth:SUMMary:PACKet:BDR:HEADer:FLOW? (on page 1611)	Returns the value of Flow field.

Command	Description
READ:BLUEtooth:SUMMary:PACKet:BDR:HEADer:HEC? (on page 1612)	Returns the value of HEC field.
READ:BLUEtooth:SUMMary:PACKet:BDR:HEADer:LT:ADDRess? (on page 1612)	Returns the LT address.
READ:BLUEtooth:SUMMary:PACKet:BDR:HEADer:SEQN? (on page 1613)	Returns the value of SEQN field.
READ:BLUEtooth:SUMMary:PACKet:BDR:HEADer:TYPE? (on page 1614)	Returns the value of Type field.
READ:BLUEtooth:SUMMary:PACKet:BDR:PAYLoad:LENgth? (on page 1614)	Returns the preamble of the Basic Rate payload length.
READ:BLUEtooth:SUMMary:PACKet:BDR:PREAmble? (on page 1615)	Returns the preamble of the Basic Rate packet.
READ:BLUEtooth:SUMMary:PACKet:BDR:SYNC:WORD? (on page 1615)	Returns the sync word.
READ:BLUEtooth:SUMMary:PACKet:BDR:TYPE? (on page 1615)	Returns the packet type of the Basic Rate signal.
READ:BLUEtooth:SUMMary:PACKet:BLE:ACCEss:ADDRess? (on page 1616)	Returns the 32-bit Access Address.
READ:BLUEtooth:SUMMary:PACKet:BLE:CRC? (on page 1616)	Returns the value of the CRC field.
READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:LENGth? (on page 1617)	Returns the value of the PDU length field.
READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:RX:ADDRess? (on page 1619)	Returns the PDU Rx address.
READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:TX:ADDRess? (on page 1620)	Returns the PDU Tx address.
READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:TYPE? (on page 1620)	Returns the PDU packet type.
READ:BLUEtooth:SUMMary:PACKet:BLE:PREAmble? (on page 1621)	Returns the preamble of the BLE packet.
READ:BLUEtooth:SUMMary:PACKet:BLE:TYPE? (on page 1621)	Returns the packet type.
READ:NOISe:TABLe:VALue? (on page 1684)	Returns the noise table data.
READ:OBWidth:BOBW:XDBBandwidth? (on page 1684)	Returns the difference between the higher and lower frequency points corresponding to the value where it is X dB less from the value at the center frequency.

Command	Description
READ:OBWidth:BOBW:XDBMbandwidth:IN:BANDwidth? (on page 1685)	Returns the difference between the higher and lower frequency points corresponding to the value which it is the X dBm value measured inwards (coming from the ends to the center frequency).
READ:OBWidth:BOBW:XDBMbandwidth:IN:LEFT:FREQuency? (on page 1686)	Returns the frequency corresponding to the X dBm value to the left of the center frequency measured in the inward direction.
READ:OBWidth:BOBW:XDBMbandwidth:IN:RIGHt:FREQuency? (on page 1686)	Returns the frequency corresponding to the X dBm value to the right of the center frequency measured in the inward direction.
READ:OBWidth:BOBW:XDBMbandwidth:OUT:BANDwidth? (on page 1687)	Returns the difference between the higher and lower frequency points corresponding to the value which it is the X dBm value measured outwards (from the center frequency to the ends).
READ:OBWidth:BOBW:XDBMbandwidth:OUT:LEFT:FREQuency? (on page 1687)	Returns the frequency corresponding to the X dBm value to the left of the center frequency measured in the outward direction.
READ:OBWidth:BOBW:XDBMbandwidth:OUT:RIGHt:FREQuency? (on page 1688)	Returns the frequency corresponding to the X dBm value to the right of the center frequency measured in the outward direction.
READ:P25 subgroup	**P25 measurements**
READ:P25:CONSte:FERRor? (on page 1718)	Returns the frequency error.
READ:P25:SUMMary:LIMIt:SRA (on page 1722)	Returns Limit of Symbol Rate Accuracy measurement results.
READ:P25:SUMMary:MODUlation:MEASurement:FIDelity:STATus (on page 1723)	Returns the Modulation Fidelity measurement status.
READ:P25:SUMMary:MODUlation:MEASurement:FIDelity? (on page 1722)	Returns the Modulation Fidelity measurement results.
READ:P25:SUMMary:MODUlation:MEASurement:FREQuency:ACCUracy? (on page 1723)	Returns the Operational Frequency Accuracy measurement results.
READ:P25:SUMMary:MODUlation:MEASurement:FREQuency:ACCUracy:STATus (on page 1724)	Returns the status of Frequency Accuracy measurement status.
READ:P25:SUMMary:MODUlation:MEASurement:FREQuency:DEVIation:NEGPeak:STATus (on page 1725)	Returns the Status of Frequency Deviation, negative peak status.
READ:P25:SUMMary:MODUlation:MEASurement:FREQuency:DEVIation:POSPeak:STATus (on page 1726)	Returns the status of Frequency Deviation, positive peak status.
READ:P25:SUMMary:MODUlation:MEASurement:FREQuency:ERRor (on page 1726)	Returns the Frequency Error measurement results.
READ:P25:SUMMary:MODUlation:MEASurement:SRA:STATus (on page 1727)	Returns Status of Symbol Rate Accuracy measurement status.
READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXOFF:STATus? (on page 1728)	Returns HCPM Pmax-off measurement status.

Command	Description
READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXOFF? (on page 1729)	This command returns the HCPM Pmax-off measurement results.
READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXON:STATus (on page 1730)	Returns Status of HCPM Pmax-on measurement status.
READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXON? (on page 1730)	This command returns the HCPM Pmax-on measurement results.
READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXSs:STATus (on page 1731)	Returns HCPM Pss-max measurement status.
READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXSs? (on page 1731)	Returns the HCPM Pss-max measurement results.
READ:P25:SUMMary:POWEr:MEASurement:HCPM:MINImum:STATus? (on page 1732)	Returns status of HCPM Pss-min measurement results based on the limit.
READ:P25:SUMMary:POWEr:MEASurement:HCPM:MINImum? (on page 1733)	Returns the HCPM Pss-min measurement results.
READ:P25:SUMMary:POWEr:MEASurement:HCPM:OFFSlot:STATus? (on page 1733)	Returns Status of HCPM Offslot measurement based on the limit.
READ:P25:SUMMary:POWEr:MEASurement:HCPM:OFFSlot? (on page 1733)	Returns the HCPM Offslot power results.
READ:P25:SUMMary:POWEr:MEASurement:HCPM:PEAK:ACPR:HI (on page 1734)	Returns P_ACP_HI value which is displayed on P25 Summary display.
READ:P25:SUMMary:POWEr:MEASurement:HCPM:PEAK:ACPR:LOW (on page 1734)	Returns P_ACP_LOW value which is displayed on P25 Summary display.
READ:P25:SUMMary:POWEr:MEASurement:HCPM:PEAK:ACPR:MIN (on page 1735)	Returns Min Pk ACPR value which is displayed on P25 Summary Display.
READ:P25:SUMMary:POWEr:MEASurement:HCPM:PEAK:ACPR:MIN:STATus (on page 1735)	Returns status of Min Pk ACPR value based on the limit.
READ:P25:SUMMary:POWEr:MEASurement:RF:STATus (on page 1736)	Returns status of RF measurement results.
READ:P25:SUMMary:POWEr:MEASurement:RF? (on page 1736)	Returns the RF output power results.
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ONE:ACQ:COUNT? (on page 1737)	Returns the HCPM Time alignment's t_error_1 acquisition count over which the t_error_1 averaging is done.
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ONE:STATus (on page 1737)	Returns status of HCPM Time alignment's t_error_1 measurement results based on the limit.

Command	Description
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ONE? (on page 1738)	Returns the HCPM Time alignment t_error_1 measurement results.
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ZERO:ACQ:COUNT? (on page 1739)	Returns the HCPM Time alignment's t_error_0 acquisition count over which the t_error_0 averaging is done.
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ZERO:STATus (on page 1739)	Returns status of HCPM Time alignment's t_error_0 measurement results.
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ZERO? (on page 1739)	Returns the HCPM Time alignment's t_error_0 measurement results over which the t_error_0 averaging is done.
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TOBSync:ACQ:COUNT? (on page 1742)	Returns the HCPM Time alignment's t_obsync acquisition counts over which the t_obsync averaging is done.
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TOBSync? (on page 1740)	Returns the HCPM Time alignment's t_obsync measurement results.
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ACQ:COUNT:BI? (on page 1742)	Returns the Average Transmitter Power Attack time busy/idle acquisition count over which the Phase 1 Attack time averaging is done.
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ACQ:COUNT? (on page 1740), READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ACQ:COUNT? (on page 1741)	Returns the Average Transmitter Power Attack acquisition count over which the Phase 1 Attack time averaging is done.
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ENCOder:BI:STATus (on page 1742)	Returns status of Average Transmitter Power Encoder Attack Time measurement results.
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ENCOder:BI?	Returns the Average Transmitter Power Encoder Attack Time busy/idle measurement results.
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ENCOder:STATus (on page 1743)	Returns Status of Average Transmitter Encoder Attack Time measurement results based on the limit.
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ENCOder? (on page 1744)	Returns the Average Transmitter Encoder Attack Time measurement results.
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr:BI:STATus (on page 1745)	Returns the Average Transmitter Encoder Attack Time measurement results.
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr:BI? (on page 1745)	Returns the Average Transmitter Power Attack Time, busy/idle measurement results.

Command	Description
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr:STATus (on page 1746)	Returns Status of Average Transmitter Power Attack Time measurement results.
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr? (on page 1746)	Returns the Average Transmitter Power Attack Time measurement results.
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy:ACQ:COUNT? (on page 1748)	Returns the Throughput delay acquisition counts over which the Throughput delay averaging is done.
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy:STATus (on page 1748)	Returns Status of Throughput delay measurement results based on the limit.
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy? (on page 1747)	Returns the Throughput delay measurement results.
READ:P25:EDIagram:FDEViation? (on page 1718)	Returns the Frequency Deviation vs Time trace data with the X values.
READ:P25:EDIagram:FERRor? (on page 1719)	Returns the frequency error.
READ:P25:PVTime:BURSt:POWer? (on page 1719)	Returns the average power of the burst packet, in dBm, for the trace.
READ:P25:PVTime:FALL:TIMe? (on page 1720)	Queries the value of the width of the falling edge, in seconds.
READ:P25:PVTime:FTIMe? (on page 1720)	Queries the value of the width of the falling edge, in seconds.
READ:P25:PVTime:TRACe:X? (on page 1721)	Returns the horizontal values (time in seconds) for the P25 Power vs. Time trace.
READ:P25:PVTime:TRACe:XY? (on page 1721)	Returns the horizontal value (time in seconds) and vertical value (power) for the P25 Power vs. Time trace <x(n)><y(n)> is the horizontal value and vertical value pair at the *n*th data point.
READ:P25:SUMMary:MODUlation:MEASurement:FIDelity (on page 1722)	Returns the Modulation Fidelity measurement results.
READ:P25:SUMMary:MODUlation:MEASurement:FREQuency:ACCUracy (on page 1723)	Returns the Operational Frequency Accuracy measurement results.
READ:P25:SUMMary:MODUlation:MEASurement:FREQuency:DEVIation:NEGPeak (on page 1724)	Returns the Frequency Deviation, negative peak, measurement results.
READ:P25:SUMMary:MODUlation:MEASurement:FREQuency:DEVIation:POSPeak (on page 1725)	Returns the Frequency Deviation, positive peak, measurement results.
READ:P25:SUMMary:MODUlation:MEASurement:FREQuency:ERRor (on page 1726)	Returns the Frequency Error measurement results.
READ:P25:SUMMary:MODUlation:MEASurement:SRA (on page 1727)	Returns the Symbol Rate Accuracy measurement results.

Command	Description
READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXOFF (on page 1729)	This command returns the HCPM Pmax-off measurement results.
READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXON (on page 1730)	This command returns the HCPM Pmax-on measurement results.
READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXSs (on page 1730)	Returns the HCPM Pss-max measurement results.
READ:P25:SUMMary:POWEr:MEASurement:HCPM:MINImum (on page 1733)	Returns the HCPM Pss-min measurement results.
READ:P25:SUMMary:POWEr:MEASurement:HCPM:OFFSlot (on page 1733)	Returns the HCPM Offslot power results.
READ:P25:SUMMary:POWEr:MEASurement:RF (on page 1736)	Returns the RF output power results.
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ONE (on page 1736)	Returns the HCPM Time alignment t_error_1 measurement results.
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ONE:ACQ:COUNT (on page 1737)	Returns the HCPM Time alignment's t_error_1 acquisition count over which the t_error_1 averaging is done.
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ZERO (on page 1739)	Returns the HCPM Time alignment's t_error_0 measurement results over which the t_error_0 averaging is done.
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ZERO:ACQ:COUNT (on page 1739)	Returns the HCPM Time alignment's t_error_0 acquisition count over which the t_error_0 averaging is done.
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TOBSync (on page 1740)	Returns the HCPM Time alignment's t_obsync measurement results.
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TOBSync:ACQ:COUNT (on page 1742)	Returns the HCPM Time alignment's t_obsync acquisition counts over which the t_obsync averaging is done.
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ACQ:COUNT (on page 1740), READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ACQ:COUNT (on page 1741)	Returns the Average Transmitter Power Attack acquisition count over which the Phase 1 Attack time averaging is done.
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ACQ:COUNT:BI (on page 1743)	Returns the Average Transmitter Power Attack time busy/idle acquisition count over which the Phase 1 Attack time averaging is done.
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ENCOder (on page 1744)	Returns the Average Transmitter Encoder Attack Time measurement results.

Command	Description
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ENCOder:BI	Returns the Average Transmitter Power Encoder Attack Time, busy/idle measurement results.
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr (on page 1744)	Returns the Average Transmitter Power Attack Time measurement results.
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr:BI (on page 1745)	Returns the Average Transmitter Power Attack Time, busy/idle measurement results.
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy (on page 1748)	Returns the Throughput delay measurement results.
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy:ACQ:COUNT (on page 1747)	Returns the Throughput delay acquisition counts over which the Throughput delay averaging is done.
READ:ACPower subgroup (continued)	
READ:ACPower:CHANnel:POWer? (on page 1554)	Returns the average power of the main channel.
READ:ACPower:SPECtrum? (on page 1554)	Returns spectrum trace data of the ACPR measurement.
READ:{AM|FM|PM} subgroup (Option 21 only)	**AM/FM/PM measurement**
READ:{AM|FM|PM}? (on page 1555)	Returns the trace data.
READ:AM:AMINdex? (on page 1556)	Returns the modulation index .
READ:AM:AMNegative? (on page 1556)	Returns the negative peak modulation factor (–AM).
READ:AM:AMPositive? (on page 1556)	Returns the positive peak modulation factor (+AM).
READ:AM:RESult? (on page 1557)	Returns the AM measurement results.
READ:FM:PHALf? (on page 1650)	Returns the half peak-peak frequency deviation (Pk-Pk/2).
READ:{FM|PM}:FERRor? (on page 1649)	Returns the frequency error in the Frequency modulation and Phase modulation measurements.
READ:FM:PNEGative? (on page 1650)	Returns the negative peak frequency deviation (–Pk).
READ:FM:PPOSitive? (on page 1651)	Returns the positive peak frequency deviation (+Pk).
READ:FM:PTPeak? (on page 1651)	Returns the peak-peak frequency deviation (Pk-Pk).
READ:FM:RESult? (on page 1652)	Returns the FM measurement results.
READ:FM:RMS? (on page 1652)	Returns the RMS frequency deviation.
READ:PM:PNEGative? (on page 1756)	Returns the negative peak phase deviation (–Pk).
READ:PM:PPOSitive? (on page 1757)	Returns the positive peak phase deviation (+Pk).

Command	Description
READ:PM:PTPeak? (on page 1757)	Returns the peak-peak phase deviation (Pk-Pk).
READ:PM:RESult? (on page 1758)	Returns the PM measurement results.
READ:PM:RMS? (on page 1758)	Returns the RMS phase deviation.
READ:AUDio subgroup	**Audio measurement**
READ:AUDio:FERRor? (on page 1557)	Returns the carrier frequency error in the audio measurement.
READ:AUDio:FREQuency? (on page 1558)	Returns the audio frequency.
READ:AUDio:HARMonic:COUNt? (on page 1558)	Returns the number of harmonics in the audio spectrum measurement.
READ:AUDio:HARMonic<x>:AMPLitude? (on page 1559)	Returns the amplitude of the specified harmonic in the audio spectrum measurement.
READ:AUDio:HARMonic<x>:FREQuency? (on page 1559)	Returns the frequency of the specified harmonic in the audio spectrum measurement.
READ:AUDio:HNOise? (on page 1560)	Returns the difference between the current RMS modulation value and the reference value.
READ:AUDio:HNREFerence? (on page 1561)	Returns the Hum and Noise RMS modulation reference value.
READ:AUDio:HPTPeak? (on page 1561)	Returns the half peak-to-peak modulation excursion.
READ:AUDio:MODDist? (on page 1562)	Returns the modulation distortion for the audio measurement.
READ:AUDio:NHARmonic:COUNt? (on page 1562)	Returns the number of non-harmonics in the audio spectrum measurement.
READ:AUDio:NHARmonic<x>:AMPLitude? (on page 1563)	Returns the amplitude of the specified non-harmonic in the audio spectrum measurement.
READ:AUDio:NHARmonic<x>:FREQuency? (on page 1563)	Returns the frequency of the specified non-harmonic in the audio spectrum measurement.
READ:AUDio:PNEGative? (on page 1564)	Returns the minus peak modulation excursion.
READ:AUDio:POWer? (on page 1564)	Returns the carrier power or signal power depending the signal type.
READ:AUDio:PPOSitive? (on page 1565)	Returns the positive peak modulation excursion.
READ:AUDio:RMS? (on page 1565)	Returns the RMS modulation excursion for the audio measurement.
READ:AUDio:SINad? (on page 1566)	Returns the signal-to-noise and distortion for the audio measurement.
READ:AUDio:SNOise? (on page 1566)	Returns the signal level to noise level with the harmonic distortion and non-harmonic distortion components removed.

Command	Description
READ:AUDio:SPECtrum:TRACe<x>? (on page 1567)	Returns the audio spectrum trace data for the audio measurement.
READ:AUDio:THDist:DB? (on page 1568)	Returns the total harmonic distortion in dB in the audio measurement.
READ:AUDio:THDist? (on page 1567)	Returns the percentage of the total harmonic distortion in the audio measurement.
READ:AUDio:TNHDist:DB? (on page 1569)	Returns the total non-harmonic distortion in dB in the audio measurement.
READ:AUDio:TNHDist? (on page 1568)	Returns the percentage of the total non-harmonic distortion in the audio measurement.
READ:AVTime subgroup	**Amplitude versus Time measurement**
READ:AVTime[:MEASview<y>]:AVERage? (on page 1569)	Returns the RMS value.
READ:AVTime[:MEASview<y>]:{FIRSt|SECond|THIRd|FOURth}? (on page 1570)	Returns the trace data.
READ:AVTime[:MEASview<y>]:MAXimum? (on page 1574)	Returns the maximum value.
READ:AVTime[:MEASview<y>]:MAXLocation?	Returns the time at the maximum.
READ:AVTime[:MEASview<y>]MINimum? (on page 1576)	Returns the minimum value.
READ:AVTime[:MEASview<y>]:MINLocation? (on page 1576)	Returns the time at the minimum.
READ:AVTime[:MEASview<y>]:OGRAm? (on page 1577)	Returns the ogram trace data.
READ:AVTime[:MEASview<y>]:RESult: {FIRSt|SECond|THIRd|FOURth}? (on page 1577)	Returns the measurement results.
READ:AVTime[:MEASview<y>]:TRACe:BITMap? (Query Only) (on page 1578)	Acquires a waveform and then returns trace waveform data of the bitmap trace in the Amplitude versus Time measurement.
READ:CCDF subgroup	**CCDF measurement**
READ:CCDF? (on page 1622)	Returns the CCDF measurement results.
READ:CCDF:{FIRSt|SECond|THIRd}:X? (on page 1623)	Returns the horizontal values of the specified trace.
READ:CCDF:{FIRSt|SECond|THIRd}:XY? (on page 1623)	Returns the horizontal and vertical value pairs of the specified trace.
READ:CCDF:{FIRSt|SECond|THIRd}[:Y]? (on page 1624)	Returns the vertical values of the specified trace.

Command	Description
READ:CONSte subgroup (Option 21 only)	**Constellation measurement**
READ:CONSte[:MEASview<x>]:FERRor? (on page 1625)	Returns the frequency error in Hz. The frequency error is the difference between the measured carrier frequency of the signal and the user selected center frequency of the analyzer.
READ:CONSte[:MEASview<x>]:RESults? (on page 1625)	Returns the constellation measurement results of EVM RMS, peak and location displayed on the bottom of the screen.
READ:CONSte[:MEASview<x>]:TRACe? (on page 1626)	Returns the constellation trace data.
READ:DDEMod subgroup (Option 21 only)	**General purpose digital modulation measurements**
READ:DDEMod[:MEASview<y>]:STABle? (on page 1627)	Returns the symbol table data.
READ:DIQVtime subgroup (Option 21 only)	**Demodulated I&Q versus Time measurement**
READ:DIQVtime[:MEASview<y>]:FERRor? (on page 1628)	Returns the frequency error.
READ:DIQVtime[:MEASview<y>]:I? (on page 1628)	Returns the I versus Time trace data.
READ:DIQVtime[:MEASview<y>]:Q? (on page 1629)	Returns the Q versus Time trace data.
READ:DPX subgroup	**DPX Spectrum measurement**
READ:DPX:DDENsity? (on page 1629)	Returns the calculated (DPX) density for the selected region.
READ:DPX:DGRam:LINE:COUNt? (on page 1630)	Returns the number of lines in the DPXogram measurement.
READ:DPX:DGRam:TIME[:SCALe]:OFFSet? (on page 1630)	Returns the time scale offset in the DPXogram measurement.
READ:DPX:RESults:TRACe<x>? (on page 1631)	Returns waveform data of specified trace.
READ:DPX:TRACe:AVERage? (on page 1631)	Returns waveform data of the average trace.
READ:DPX:TRACe:BITMap? (on page 1632)	Returns trace waveform data of the bitmap trace.
READ:DPX:TRACe:MATH? (on page 1632)	Returns waveform data of the math trace.
READ:DPX:TRACe:MAXimum? (on page 1633)	Returns waveform data of the maximum trace.
READ:DPX:TRACe:MINimum? (on page 1634)	Returns waveform data of the minimum trace.
READ:EDIagram subgroup (Option 21 only)	**Eye diagram**
READ:EDIagram[:MEASview<y>]:FDEViation? (on page 1634)	Returns the frequency deviation versus Time trace data with the X values.

Command	Description
READ:EDIagram[:MEASview<y>]:FERRor? (on page 1635)	Returns the frequency error in the eye diagram measurement.
READ:EDIagram[:MEASview<y>]:I? (on page 1635)	Returns the I versus Time trace data.
READ:EDIagram[:MEASview<y>]:Q? (on page 1636)	Returns the Q versus Time trace data.
READ:EVM subgroup (Option 21 only)	**EVM versus Time measurement**
READ:EVM[:MEASview<y>]:FERRor? (on page 1645)	Returns the frequency error in the EVM versus Time measurement.
READ:EVM[:MEASview<y>]:PEAK? (on page 1646)	Returns the peak value.
READ:EVM[:MEASview<y>]:PINDex? (on page 1646)	Returns the time at the EVM peak.
READ:EVM[:MEASview<y>]:RMS? (on page 1647)	Returns the RMS value.
READ:EVM[:MEASview<y>]:TRACe? (on page 1648)	Returns the EVM versus Time trace data.
READ:FDVTime subgroup (Option 21 only)	**Frequency deviation versus Time measurement**
READ:FDVTime[:MEASview<y>]:FERRor? (on page 1648)	Returns the frequency error in the Frequency deviation versus Time measurement.
READ:FDVTime[:MEASview<y>]:TRACe? (on page 1649)	Returns the Frequency deviation versus Time trace data.
READ:{FSETtling|PSETtling} subgroup	**Frequency and Phase Settling Time measurements**
READ:FSETtling:ERRor? (on page 1659)	Returns the settled error in Hz in Frequency Settling.
READ:{FSETtling|PSETtling}:FTTime? (on page 1653)	Returns the settling time from the trigger position in seconds.
READ:{FSETtling|PSETtling}:MASK[:PASS]? (on page 1653)	Returns whether the input signal passes the mask test.
READ:{FSETtling|PSETtling}:SETTled:FREQuency? (on page 1654)	Returns the frequency at which the signal is considered settled.
READ:{FSETtling|PSETtling}:SETTled[:PASS]? (on page 1654)	Returns whether the input signal is settled with the tolerance range.
READ:{FSETtling|PSETtling}:SETTled:TIME? (on page 1655)	Returns the settled time in seconds.
READ:{FSETtling|PSETtling}:SLMSd[:PASS]? (on page 1655)	Returns whether the input signal is settled within the specified tolerance and the signal is settled longer than the Minimum Settled Duration.

Command	Description
READ:{FSETtling|PSETtling}:STARt:TIME? (on page 1655)	Returns the start time for the measurement in seconds.
READ:{FSETtling|PSETtling}:TIME? (on page 1656)	Returns the settling time in seconds.
READ:{FSETtling|PSETtling}:TRACe<x>:X? (on page 1656)	Returns the Frequency or Phase values of the specified trace.
READ:{FSETtling|PSETtling}:TRACe<x>:XY? (on page 1657)	Returns the time and frequency or phase value pairs of the specified trace.
READ:{FSETtling|PSETtling}:TRACe<x>[:Y]? (on page 1658)	Returns the frequency or phase values of the specified trace.
READ:{FSETtling|PSETtling}:TRIGger:TIME? (on page 1658)	Returns the time when the trigger occurred in seconds.
READ:{FSETtling|PSETtling}:VALue? (on page 1659)	Returns the settled value in Hz for Frequency Settling and in degrees for Phase Settling.
READ:FVTime subgroup	**Frequency versus Time measurement**
READ:FVTime[:MEASview<y>]? (on page 1660)	Returns the Frequency versus Time trace data.
READ:FVTime[:MEASview<y>]:MAXimum? (on page 1660)	Returns the maximum value.
READ:FVTime[:MEASview<y>]:MAXLocation? (on page 1661)	Returns the time at which the frequency drift is maximum.
READ:FVTime[:MEASview<y>]:MINimum? (on page 1661)	Returns the minimum value.
READ:FVTime[:MEASview<y>]:MINLocation? (on page 1662)	Returns the time at which the frequency drift is minimum.
READ:FVTime[:MEASview<y>]:OGRAm? (Query only) (on page 1662)	Returns the ogram trace data.
READ:FVTime[:MEASview<y>]:RESult? (on page 1663)	Returns the measurement results.
READ:FVTime[:MEASview<y>]:TRACe:BITMap? (Query Only) (on page 1664)	Acquires a waveform and then returns trace waveform data of the bitmap trace in the Frequency versus Time measurement.
READ:IQVTime subgroup	**RF I&Q versus Time measurement**
READ:IQVTime[:MEASview<y>]:I? (on page 1664)	Returns the I versus Time trace data.
READ:IQVTime[:MEASview<y>]:MAXimum? (on page 1665)	Returns the maximum value.

Command	Description
READ:IQVTime[:MEASview<y>]:MAXLocation? (on page 1665)	Returns the time at which the I or Q level is maximum.
READ:IQVTime[:MEASview<y>]:MINimum? (on page 1666)	Returns the minimum value.
READ:IQVTime[:MEASview<y>]:MINLocation? (on page 1666)	Returns the time at which the I or Q level is minimum.
READ:IQVTime[:MEASview<y>]:Q? (on page 1667)	Returns the Q versus Time trace data.
READ:IQVTime[:MEASview<y>]:RESult? (on page 1667)	Returns the measurement results.
READ:MCPower subgroup	**MCPR measurement**
READ:MCPower:ADJacent:CHANnels? (on page 1678)	Returns the power of adjacent channels.
READ:MCPower:CHANnel:POWer? (on page 1679)	Returns the reference power.
READ:MCPower:MAIN:CHANnels? (on page 1680)	Returns the power of main channels.
READ:MCPower:SPECtrum? (on page 1680)	Returns spectrum trace data.
READ:MERRor subgroup (Option 21 only)	**Magnitude Error versus Time measurement**
READ:MERRor[:MEASview<y>]:FERRor? (on page 1681)	Returns the frequency error.
READ:MERRor[:MEASview<y>]:PEAK? (on page 1682)	Returns the peak value.
READ:MERRor[:MEASview<y>]:PINDex? (on page 1682)	Returns the time at the Magnitude error peak.
READ:MERRor[:MEASview<y>]:RMS? (on page 1683)	Returns the RMS (Root-Mean-Square) value.
READ:MERRor[:MEASview<y>]:TRACe? (on page 1683)	Returns the Magnitude error versus Time trace data.
READ:OBWidth subgroup	**Occupied Bandwidth measurement**
READ:OBWidth:FREQuency:ERRor? (on page 1688)	Returns the frequency error.
READ:OBWidth:OBWidth:BANDwidth? (on page 1689)	Returns the occupied bandwidth.
READ:OBWidth:OBWidth:LEFT:FREQuency? (on page 1689)	Returns the left (lower) frequency of the occupied bandwidth.

Command	Description
READ:OBWidth:OBWidth:LEFT:LEVel? (on page 1690)	Returns the level at the left frequency of the occupied bandwidth.
READ:OBWidth:OBWidth:POWer? (on page 1690)	Returns the reference power in the Occupied Bandwidth measurement.
READ:OBWidth:OBWidth:RIGHt:FREQuency? (on page 1691)	Returns the right (higher) frequency of the occupied bandwidth.
READ:OBWidth:OBWidth:RIGHt:LEVel? (on page 1691)	Returns the level at the right frequency of the occupied bandwidth.
READ:OBWidth:SPECtrum? (on page 1692)	Returns spectrum trace data of the Occupied Bandwidth measurement.
READ:OBWidth:XDBBandwidth:BANDwidth? (on page 1692)	Returns the x dB bandwidth.
READ:OBWidth:XDBBandwidth:LEFT:FREQuency? (on page 1693)	Returns the left (lower) frequency of the x dB bandwidth.
READ:OBWidth:XDBBandwidth:LEFT:LEVel? (on page 1693)	Returns the level at the left frequency of the x dB bandwidth.
READ:OBWidth:XDBBandwidth:POWer? (on page 1694)	Returns the reference power in the x dB bandwidth measurement.
READ:OBWidth:XDBBandwidth:RIGHt:FREQuency? (on page 1694)	Returns the right (higher) frequency of the x dB bandwidth.
READ:OBWidth:XDBBandwidth:RIGHt:LEVel? (on page 1695)	Returns the level at the right frequency of the x dB bandwidth.
READ:OFDM subgroup	**Orthogonal Frequency Division Multiplexing measurement**
READ:OFDM:APOWer? (on page 1695)	Returns the average power.
READ:OFDM:APOWer:PEAK? (on page 1696)	Returns the peak-to-average power.
READ:OFDM:CONSte:MAGNitude? (on page 1696)	Returns the constellation magnitude data.
READ:OFDM:CONSte:PHASe? (on page 1697)	Returns the constellation phase data.
READ:OFDM:CONSte:TYPE? (on page 1698)	Returns the constellation context value
READ:OFDM:CONSte:VALue? (on page 1699)	Returns the constellation value.
READ:OFDM:CPE? (on page 1699)	Returns the Common Pilot Error magnitude.
READ:OFDM:CRESponse:MAGNitude? (on page 1700)	Returns the channel response magnitude data.
READ:OFDM:CRESponse:PHASe? (on page 1700)	Returns the channel response phase data.

Command	Description
READ:OFDM:EVM:PEAK:DECibel:ALL? (on page 1701)	Returns the peak EVM data for all subcarriers.
READ:OFDM:EVM:PEAK:DECibel:DATA? (on page 1701)	Returns the peak EVM data for the data subcarriers.
READ:OFDM:EVM:PEAK:DECibel:PILots? (on page 1702)	Returns the peak EVM data for the pilot subcarriers.
READ:OFDM:EVM:PEAK:PERCent:ALL? (on page 1702)	Returns the peak EVM data for all subcarriers.
READ:OFDM:EVM:PEAK:PERCent:DATA? (on page 1703)	Returns the peak EVM data for the data subcarriers.
READ:OFDM:EVM:PEAK:PERCent:PILots? (on page 1703)	Returns the peak EVM data for the pilot subcarriers.
READ:OFDM:EVM:PEAK:SCARrier:ALL? (on page 1704)	Returns the peak EVM data for all subcarriers.
READ:OFDM:EVM:PEAK:SCARrier:DATA? (on page 1704)	Returns the peak EVM data for the data subcarriers.
READ:OFDM:EVM:PEAK:SCARrier:PILots? (on page 1704)	Returns the peak EVM data for the pilot subcarriers.
READ:OFDM:EVM:PEAK:SYMBol:ALL? (on page 1705)	Returns the peak EVM data for all subcarriers.
READ:OFDM:EVM:PEAK:SYMBol:DATA? (on page 1705)	Returns the peak EVM data for the data subcarriers
READ:OFDM:EVM:PEAK:SYMBol:PILots? (on page 1706)	Returns the peak EVM data for the pilot subcarriers.
READ:OFDM:EVM:RMS:DECibel:ALL? (on page 1706)	Returns the RMS EVM data for all subcarriers.
READ:OFDM:EVM:RMS:DECibel:DATA? (on page 1707)	Returns the RMS EVM data for the data subcarriers.
READ:OFDM:EVM:RMS:DECibel:PILots? (on page 1707)	Returns the RMS EVM data for the pilot subcarriers.
READ:OFDM:EVM:RMS:PERCent:ALL? (on page 1708)	Returns the peak RMS data for all subcarriers.
READ:OFDM:EVM:RMS:PERCent:DATA? (on page 1708)	Returns the peak RMS data for the data subcarriers.
READ:OFDM:EVM:RMS:PERCent:PILots? (on page 1708)	Returns the peak RMS data for the pilot subcarriers.

Command	Description
READ:OFDM:EVM:TRACe<x>? (on page 1709)	Returns the EVM trace data.
READ:OFDM:FERRor? (on page 1710)	Returns the Frequency error reading.
READ:OFDM:FLATness:PASS? (on page 1710)	Verifies the average power levels of the subcarriers remain within the limits defined for a particular standard.
READ:OFDM:FLATness:RESult? (on page 1710)	Returns the summary results of each segment.
READ:OFDM:FLATness:TRACe<x>? (on page 1711)	Returns the OFDM Spectral Flatness trace data.
READ:OFDM:GAIN:IMBalance? (on page 1712)	Returns the gain imbalance.
READ:OFDM:IQ:ORIGin:OFFSet? (on page 1712)	Returns the IQ origin offset.
READ:OFDM:MERRor:TRACe<x>? (on page 1713)	Returns the magnitude error trace data.
READ:OFDM:PACKet:DIRection? (on page 1713)	Returns the direction of the packet.
READ:OFDM:PERRor:TRACe<x>? (on page 1714)	Returns the phase error trace data.
READ:OFDM:POWer:TRACe<x>? (on page 1714)	Returns the power trace data.
READ:OFDM:QUADrature:OFFSet? (on page 1715)	Returns the quadrature offset.
READ:OFDM:SCARriers? (on page 1716)	Returns the number of subcarriers.
READ:OFDM:SCARriers:SPACing? (on page 1716)	Returns the subcarrier spacing.
READ:OFDM:STABle:VALUe? (on page 1716)	Returns the number of bytes in the symbol table.
READ:OFDM:SYMBol:CERRor? (on page 1717)	Returns the symbol clock error.
READ:OFDM:SYMBol:COUNt? (on page 1717)	Returns the number of symbols.
READ:PERRor subgroup (Option 21 only)	**Phase error versus Time measurement**
READ:PERRor[:MEASview<y>]:FERRor? (on page 1749)	Returns the frequency error in the Phase error versus Time measurement.
READ:PERRor[:MEASview<y>]:PEAK? (on page 1749)	Returns the peak value in the Phase error versus Time measurement.
READ:PERRor[:MEASview<y>]:PINDex? (on page 1750)	Returns the time at the phase error peak.
READ:PERRor[:MEASview<y>]:RMS (on page 1751)	Returns the RMS (Root-Mean-Square) value in the Phase error versus Time measurement.

Command	Description
READ:PERRor[:MEASview<y>]:TRACe? (on page 1751)	Returns the Phase error versus Time trace data.
READ:PHVTime subgroup	**Phase versus Time measurement**
READ:PHVTime[:MEASview<y>]? (on page 1752)	Returns the Phase versus Time trace data.
READ:PHVTime[:MEASview<y>]:MAXimum? (on page 1752)	Returns the maximum value.
READ:PHVTime[:MEASview<y>]:MAXLocation? (on page 1753)	Returns the time at which the phase is maximum.
READ:PHVTime[:MEASview<y>]:MINimum? (on page 1753)	Returns the minimum value.
READ:PHVTime[:MEASview<y>]:MINLocation? (on page 1754)	Returns the time at which the phase is minimum.
READ:PHVTime[:MEASview<y>]:OGRAm? (Query Only) (on page 1754)	Returns the ogram trace data.
READ:PHVTime[:MEASview<y>]:RESult? (on page 1755)	Returns the results.
READ:PHVTime[:MEASview<y>]:TRACe:BITMap? (Query Only) (on page 1756)	Acquires a waveform and then returns trace waveform data of the bitmap trace in the Phase versus Time measurement.
READ:PNOise subgroup (Option 11 only)	**Phase noise measurements**
READ:PNOise:ALL? (on page 1759)	Returns all the measurement results.
READ:PNOise:CARRier:FERRor? (on page 1759)	Returns the carrier frequency error.
READ:PNOise:CARRier:POWer? (on page 1760)	Returns the carrier power.
READ:PNOise:RESidual:FM? (on page 1761)	Returns the residual FM.
READ:PNOise:RMS:PNOise? (on page 1761)	Returns the RMS phase noise.
READ:PNOise:SPECtrum<x>:X? (on page 1762)	Returns the frequencies of the specified trace.
READ:PNOise:SPECtrum<x>:XY? (on page 1763)	Returns the frequency and phase noise pairs of the specified trace.
READ:PNOise:SPECtrum<x>[:Y]? (on page 1762)	Returns the phase noise values of the specified trace.
READ:PNOise:JITTer? (on page 1760)	Returns the jitter.
READ:PULSe subgroup	**Pulsed RF measurements**
READ:PULSe[:RESult][:MEASview<y>]:ATX? (on page 1773)	Returns the average transmitted power in the results table.

Command	Description
READ:PULSe[:RESult][:MEASview<y>]:AVERage? (on page 1774)	Returns the average on power in the results table.
READ:PULSe[:RESult][:MEASview<y>]:DRODb? (on page 1774)	Returns the droop in dB in the pulse table.
READ:PULSe[:RESult][:MEASview<y>]:DROop? (on page 1775)	Returns the droop in the pulse table.
READ:PULSe[:RESult][:MEASview<y>]:DUTPct? (on page 1775)	Returns the duty factor (%) in the results table.
READ:PULSe[:RESult][:MEASview<y>]:DUTRatio? (on page 1776)	Returns the duty factor (ratio) in the results table.
READ:PULSe[:RESult][:MEASview<y>]:FALL? (on page 1777)	Returns the fall time in the results table.
READ:PULSe[:RESult][:MEASview<y>]:FDELta? (on page 1778)	Returns the delta frequency in the results table.
READ:PULSe[:RESult][:MEASview<y>]:FRDeviation? (on page 1778)	Returns the frequency deviation in the results table.
READ:PULSe[:RESult][:MEASview<y>]:IRAMplitude? (on page 1779)	Returns the pulse impulse response amplitude in the pulse table.
READ:PULSe[:RESult][:MEASview<y>]:IRTime? (on page 1779)	Returns the impulse response time in the pulse table.
READ:PULSe[:RESult][:MEASview<y>]:MFReqerror? (on page 1780)	Returns the maximum frequency error in the results table.
READ:PULSe[:RESult][:MEASview<y>]:MPHerror? (on page 1780)	Returns the maximum phase error in the results table.
READ:PULSe[:RESult][:MEASview<y>]:OVEDb? (on page 1781)	Returns the overshoot in dB in the pulse table.
READ:PULSe[:RESult][:MEASview<y>]:OVERshoot? (on page 1782)	Returns the overshoot in the pulse table.
READ:PULSe[:RESult][:MEASview<y>]:PHDeviation? (on page 1782)	Returns the phase deviation in the results table.
READ:PULSe[:RESult][:MEASview<y>]:PPFRequency? (on page 1783)	Returns the pulse-pulse carrier frequency in the results table.
READ:PULSe[:RESult][:MEASview<y>]:PPOWer? (on page 1784)	Returns the peak power in the results table.
READ:PULSe[:RESult][:MEASview<y>]:PPPHase? (on page 1785)	Returns the pulse-pulse carrier phase in the results table.

Command	Description
READ:PULSe[:RESult][:MEASview<y>]:RINTerval? (on page 1786)	Returns the repetition interval in the results table.
READ:PULSe[:RESult][:MEASview<y>]:RIPDb? (on page 1786)	Returns the ripple in dB in the pulse table.
READ:PULSe[:RESult][:MEASview<y>]:RIPPle? (on page 1787)	Returns the ripple in the results table.
READ:PULSe[:RESult][:MEASview<y>]:RISE? (on page 1787)	Returns the rise time in the results table.
READ:PULSe[:RESult][:MEASview<y>]:RMSFreqerror? (on page 1788)	Returns the RMS frequency error in the results table.
READ:PULSe[:RESult][:MEASview<y>]:RMSPherror? (on page 1789)	Returns the RMS phase error in the results table.
READ:PULSe[:RESult][:MEASview<y>]:RRATe? (on page 1789)	Returns the repetition rate in the results table.
READ:PULSe[:RESult][:MEASview<y>]:TIME? (on page 1790)	Returns the time in the results table.
READ:PULSe[:RESult][:MEASview<y>]:WIDTh? (on page 1790)	Returns the pulse width in the results table.
READ:PULSe:STATistics[:MEASview<y>]? (on page 1791)	Returns the trace data of the pulse statistics measurement.
READ:PULSe:STATistics[:MEASview<y>]:ATX?	Returns the average transmitted power of the statistics.
READ:PULSe:STATistics[:MEASview<y>]:AVERage? (on page 1795)	Returns the average on power of the statistics.
READ:PULSe:STATistics[:MEASview<y>]:DRODb? (on page 1796)	Returns the droop in dB of the statistics.
READ:PULSe:STATistics[:MEASview<y>]:DROop? (on page 1796)	Returns the droop of the statistics.
READ:PULSe:STATistics[:MEASview<y>]:DUTPct? (on page 1797)	Returns the duty factor (%) of the statistics.
READ:PULSe:STATistics[:MEASview<y>]:DUTRatio? (on page 1798)	Returns the duty factor (ratio) of the statistics.
READ:PULSe:STATistics[:MEASview<y>]:FALL? (on page 1799)	Returns the fall time of the statistics.
READ:PULSe:STATistics[:MEASview<y>]:FDELta? (on page 1799)	Returns the delta frequency in the pulse statistics.

Command	Description
READ:PULSe:STATistics[:MEASview<y>]:FRDeviation? (on page 1800)	Returns the frequency deviation of the statistics.
READ:PULSe:STATistics[:MEASview<y>]:IRAMplitude? (on page 1801)	Returns the Pulse Impulse Response amplitude measurement in the pulse statistics.
READ:PULSe:STATistics[:MEASview<y>]:IRTime? (on page 1801)	Returns the impulse response time in the pulse statistics.
READ:PULSe:STATistics[:MEASview<y>]:MFReqerror? (on page 1802)	Returns the maximum frequency error of the statistics.
READ:PULSe:STATistics[:MEASview<y>]:MPHerror? (on page 1804)	Returns the maximum phase error of the statistics.
READ:PULSe:STATistics[:MEASview<y>]:OVEDb? (on page 1804)	Returns the overshoot measurement result in dB in the pulse statistics
READ:PULSe:STATistics[:MEASview<y>]:OVERshoot? (on page 1805)	Returns the overshoot measurement result in the pulse statistics
READ:PULSe:STATistics[:MEASview<y>]:PHDeviation? (on page 1806)	Returns the phase deviation of the statistics.
READ:PULSe:STATistics[:MEASview<y>]:PPFRequency? (on page 1807)	Returns the pulse-pulse carrier frequency of the statistics.
READ:PULSe:STATistics[:MEASview<y>]:PPOWer? (on page 1807)	Returns the peak power of the statistics.
READ:PULSe:STATistics[:MEASview<y>]:PPPHase? (on page 1809)	Returns the pulse-pulse carrier phase of the statistics.
READ:PULSe:STATistics[:MEASview<y>]:RINTerval? (on page 1809)	Returns the repetition interval of the statistics.
READ:PULSe:STATistics[:MEASview<y>]:RIPDb? (on page 1810)	Returns the ripple in dB of the statistics
READ:PULSe:STATistics[:MEASview<y>]:RIPPle? (on page 1811)	Returns the ripple of the statistics.
READ:PULSe:STATistics[:MEASview<y>]:RISE? (on page 1811)	Returns the rise time of the statistics.
READ:PULSe:STATistics[:MEASview<y>]:RMSFreqerror? (on page 1812)	Returns the RMS frequency error of the statistics.
READ:PULSe:STATistics[:MEASview<y>]:RMSPherror? (on page 1813)	Returns the RMS phase error of the statistics.
READ:PULSe:STATistics[:MEASview<y>]:RRATe? (on page 1813)	Returns the repetition rate of the statistics.

Command	Description
READ:PULSe:STATistics[:MEASview<y>]:WIDTh? (on page 1814)	Returns the pulse width of the statistics.
READ:PULSe:STATistics[:MEASview<y>]:X? (on page 1814)	Returns the horizontal values of the trace data of the pulse statistics measurement.
READ:PULSe:STATistics[:MEASview<y>]:XY? (on page 1815)	Returns the horizontal and vertical values of the trace data of the pulse statistics measurement.
READ:PULSe:TRACe[:MEASview<y>]:Y (on page 1816)	Returns the vertical values of the trace data of the pulse statistics measurement.
READ:PULSe:TRACe[:MEASview<y>]:X? (on page 1814)	Returns the time values of the pulse trace.
READ:PULSe:TRACe[:MEASview<y>]:XY? (on page 1815)	Returns the horizontal (time) and vertical value pairs of the pulse trace.
READ:PULSe:TRACe[:MEASview<y>]:Y? (on page 1816)	Returns the vertical values of the pulse trace.
READ:SEM subgroup	**Spectral Emissions Mask measurement**
READ:SEM:MHITs:COUNt? (on page 1818)	Returns the number of rows in the results table.
READ:SEM:MHITs<x>:AMPLitude:ABSolute? (on page 1818)	Returns the absolute amplitude of the specified mask hit signal.
READ:SEM:MHITs<x>:AMPLitude:RELative? (on page 1819)	Returns the relative amplitude of the specified mask hit signal.
READ:SEM:MHITs<x>:FREQuency? (on page 1819)	Returns the frequency of the specified mask hit signal.
READ:SEM:MHITs<x>:INTeg:ABSolute? (on page 1820)	Returns the absolute integration amplitude of the specified mask hit signal.
READ:SEM:MHITs<x>:INTeg:RELative? (on page 1820)	Returns the relative integration amplitude of the specified mask hit signal.
READ:SEM:MHITs<x>:MARGin:ABSolute? (on page 1821)	Returns the absolute margin for the specified mask hit signal.
READ:SEM:MHITs<x>:MARGin:RELative? (on page 1821)	Returns the relative margin for the specified mask hit signal.
READ:SEM:MHITs<x>:OFFSet? (on page 1822)	Returns the offset for the specified mask hit signal.
READ:SEM:PASS? (on page 1822)	Returns the pass/fail limit test result.
READ:SEM:REF:POWer? (on page 1823)	Returns the reference power level.
READ:SEM:SPECtrum:X? (on page 1823)	Returns the frequencies of the spectrum trace.

Command	Description
READ:SEM:SPECtrum:XY? (on page 1824)	Returns the frequency and amplitude pairs of the spectrum trace.
READ:SEM:SPECtrum[:Y]? (on page 1824)	Returns the amplitude of the spectrum trace.
READ:SGRam subgroup	**Spectrogram measurement**
READ:SGRam? (on page 1825)	Returns the spectrogram trace data.
READ:SPECtrum subgroup	**Spectrum measurement**
READ:SPECtrum:TRACe<x>? (on page 1825)	Returns the trace data in the Spectrum Analyzer measurement.
READ:SPECtrum[:MEASview<y>]:TRACe:BITMap? (Query Only) (on page 1826)	Acquires a waveform and then returns trace waveform data of the bitmap trace in the Spectrum measurement.
READ:SPURious subgroup	**Spurious measurement**
READ:SPURious:CARRier:POWer? (on page 1827)	Returns the carrier power.
READ:SPURious:COUNt? (on page 1827)	Returns the number of spurious signals.
READ:SPURious:PASS? (on page 1828)	Returns the pass/fail limit test result.
READ:SPURious:SPECtrum:X? (on page 1828)	Returns the frequencies of the spectrum trace.
READ:SPURious:SPECtrum:XY? (on page 1829)	Returns the frequency and amplitude pairs of the spectrum trace.
READ:SPURious:SPECtrum[:Y]? (on page 1829)	Returns the amplitudes of the spectrum trace.
READ:SPURious:SPUR<x>:AMPLitude:ABSolute? (on page 1830)	Returns the absolute amplitude of the specified spurious signal.
READ:SPURious:SPUR<x>:AMPLitude:RELative? (on page 1830)	Returns the relative amplitude of the specified spurious signal.
READ:SPURious:SPUR<x>:FREQuency:ABSolute? (on page 1831)	Returns the absolute frequency of the specified spurious signal.
READ:SPURious:SPUR<x>:FREQuency:RELative? (on page 1831)	Returns the relative frequency of the specified spurious signal.
READ:SPURious:SPUR<x>:LIMit:ABSolute? (on page 1832)	Returns the absolute amplitude of the limit for a spurious signal.
READ:SPURious:SPUR<x>:LIMit:RELative? (on page 1832)	Returns the relative amplitude of the limit for a spurious signal.
READ:SPURious:SPUR<x>:LIMit:VIOLation? (on page 1833)	Returns whether the specified spurious signal exceeds the limit or not.
READ:SPURious:SPUR<x>:RANGe? (on page 1833)	Returns the frequency range in which the spurious signal occurred.

Command	Description
READ:SQUality subgroup (Option 21 only)	**Signal quality measurement**
READ:SQUality[:MEASview<y>]:FREQuency:DEViation? (on page 1834)	Returns the frequency deviation in the signal quality measurement.
READ:SQUality[:MEASview<y>]:FREQuency:DEViation:TABLe? (on page 1834)	Returns the frequency deviation measurement results table.
READ:SQUality[:MEASview<y>]:FREQuency:ERRor? (on page 1835)	Returns the frequency error in the signal quality measurement.
READ:SQUality[:MEASview<y>]:GAIN:IMBalance? (on page 1836)	Returns the gain imbalance in the signal quality measurement.
READ:SQUality[:MEASview<y>]:ORIGin:OFFSet? (on page 1836)	Returns the origin offset in the signal quality measurement.
READ:SQUality[:MEASview<y>]:PEAK:EVM? (on page 1837)	Returns the peak EVM (%) in the signal quality measurement.
READ:SQUality[:MEASview<y>]:PEAK:EVM:DB? (on page 1837)	Returns the peak EVM (dB) in the signal quality measurement.
READ:SQUality[:MEASview<y>]:PEAK:EVM:DB:OFFSet? (on page 1838)	Returns the peak offset EVM (dB) in the signal quality measurement.
READ:SQUality[:MEASview<y>]:PEAK:EVM:LOCation? (on page 1838)	Returns the time at which the EVM is peak.
READ:SQUality[:MEASview<y>]:PEAK:EVM:LOCation:OFFSet? (on page 1839)	Returns the time at which the offset EVM is peak.
READ:SQUality[:MEASview<y>]:PEAK:EVM:OFFSet? (on page 1840)	Returns the peak offset EVM (%) in the signal quality measurement.
READ:SQUality[:MEASview<y>]:PEAK:FERRor? (on page 1840)	Returns the peak FSK error in the signal quality measurement.
READ:SQUality[:MEASview<y>]:PEAK:MERRor? (on page 1841)	Returns the peak magnitude error (%) in the signal quality measurement.
READ:SQUality[:MEASview<y>]:PEAK:MERRor:DB? (on page 1841)	Returns the peak magnitude error (dB) in the signal quality measurement.
READ:SQUality[:MEASview<y>]:PEAK:MERRor:LOCation? (on page 1842)	Returns the time at which the magnitude error is peak.
READ:SQUality[:MEASview<y>]:PEAK:PERRor? (on page 1843)	Returns the peak phase error in the signal quality measurement.
READ:SQUality[:MEASview<y>]:PEAK:PERRor:LOCation? (on page 1843)	Returns the time at which the phase error is peak.

Command	Description
READ:SQUality[:MEASview<y>]:QUADrature:ERRor? (on page 1844)	Returns the quadrature error in the signal quality measurement.
READ:SQUality[:MEASview<y>]:RHO? (on page 1844)	Returns the ρ (waveform quality) in the signal quality measurement.
READ:SQUality[:MEASview<y>]:RMS:EVM? (on page 1845)	Returns the RMS EVM (%) in the signal quality measurement.
READ:SQUality[:MEASview<y>]:RMS:EVM:DB? (on page 1845)	Returns the RMS EVM (dB) in the signal quality measurement.
READ:SQUality[:MEASview<y>]:RMS:EVM:DB:OFFSet? (on page 1846)	Returns the RMS offset EVM (dB) in the signal quality measurement.
READ:SQUality[:MEASview<y>]:RMS:EVM:OFFSet? (on page 1846)	Returns the RMS offset EVM (dB) in the signal quality measurement.
READ:SQUality[:MEASview<y>]:RMS:FERRor? (on page 1847)	Returns the RMS FSK error in the signal quality measurement.
READ:SQUality[:MEASview<y>]:RMS:MER:DB? (on page 1848)	Returns the RMS MER (dB) in the signal quality measurement.
READ:SQUality[:MEASview<y>]:RMS:MERRor? (on page 1848)	Returns the RMS magnitude error (%) in the signal quality measurement.
READ:SQUality[:MEASview<y>]:RMS:MERRor:DB? (on page 1849)	Returns the RMS magnitude error (dB) in the signal quality measurement.
READ:SQUality[:MEASview<y>]:RMS:PERRor? (on page 1849)	Returns the RMS phase error in the signal quality measurement.
READ:SQUality[:MEASview<y>]:SYMBol:LENGth? (on page 1850)	Returns the number of analyzed symbols.
READ:SQUality[:MEASview<y>]:SYMBol:RATE? (on page 1850)	Returns the calculated symbol rate in Hz.
READ:SQUality[:MEASview<y>]:SYMBol:RATE:ERRor? (on page 1851)	Returns the value of the symbol rate error in percent.
READ:TDIagram subgroup (Option 21 only)	**Trellis diagram**
READ:TDIagram[:MEASview<y>]:FERRor? (on page 1851)	Returns the frequency error.
READ:TDIagram[:MEASview<y>]:TRACe? (on page 1852)	Returns the Trellis diagram trace data.
READ:TOVerview subgroup	**Time overview**
READ:TOVerview[:MEASview<y>]? (on page 1853)	Returns the trace data.

Command	Description
READ:WLAN subgroup	**Wireless LAN measurements**
READ:WLAN:CONSte:MAGNitude? (on page 1904)	Returns the magnitude data for the WLAN Constellation measurement.
READ:WLAN:CONSte:PHASe? (on page 1904)	Returns the phase values for the WLAN Constellation measurement.
READ:WLAN:CONSte:TYPE? (on page 1905)	Returns the type (context) for the WLAN Constellation measurement.
READ:WLAN:CONSte:VALue? (on page 1905)	Returns the value (decoded symbol) for the WLAN Constellation measurement.
READ:WLAN:CRESponse:MAGNitude? (on page 1906)	Returns the magnitude data of the WLAN Channel Response measurement.
READ:WLAN:CRESponse:PHASe? (on page 1906)	Returns the phase data of the WLAN Channel Response measurement.
READ:WLAN:EVM:TRACe<x>? (on page 1907)	Returns the trace data for the selected WLAN EVM trace.
READ:WLAN:FLATness:PASS? (on page 1908)	Verifies the average power level values of the subcarriers remain within the limits defined for a particular standard.
READ:WLAN:FLATness:RESult? (on page 1908)	Returns the summary results of each segment of the WLAN Spectral Flatness display.
READ:WLAN:FLATness:TRACe<x>? (on page 1909)	Returns the WLAN Spectral Flatness trace data.
READ:WLAN:MERRor:TRACe<x>? (on page 1909)	Returns the trace data for the selected WLAN Magnitude Error trace.
READ:WLAN:PERRor:TRACe<x>? (on page 1910)	Returns the trace data for the selected WLAN Phase Error trace.
READ:WLAN:PVTime:BURSt:POWer? (on page 1910)	Returns the average power of the burst packet.
READ:WLAN:PVTime:BURSt:WIDTh? (on page 1911)	Returns the value of the measured time width of the burst packet from Power-On to Power-Down.
READ:WLAN:PVTime:FALL:TIME? (on page 1911)	Returns the width of the falling edge.
READ:WLAN:PVTime:FALL:TIME:STARt? (on page 1912)	Returns the start time of the falling edge.
READ:WLAN:PVTime:FALL:TIME:STOP? (on page 1912)	Returns the stop time of the falling edge.
READ:WLAN:PVTime:FTIMe? (on page 1912)	Returns the width of the falling edge.
READ:WLAN:PVTime:RISE:TIME? (on page 1913)	Returns the width of the rising edge.
READ:WLAN:PVTime:RISE:TIME:STARt? (on page 1913)	Returns the start time of the rising edge.

Command	Description
READ:WLAN:PVTime:RISE:TIME:STOP? (on page 1914)	Returns the stop time of the rising edge.
READ:WLAN:PVTime:RTIMe? (on page 1914)	Returns the width of the rising edge.
READ:WLAN:PVTime:TRACe:X? (on page 1914)	Returns the horizontal values.
READ:WLAN:PVTime:TRACe:XY? (on page 1915)	Returns the horizontal value (time in seconds) and vertical value (power).
READ:WLAN:PVTime:TRACe[:Y]? (on page 1916)	Returns the vertical values (power).
READ:WLAN:STABle:VALUe? (on page 1916)	Returns the number of bytes in the symbol table.
READ:WLAN:SUMMary:BURSt:INDex? (on page 1917)	Returns the index of the analyzed packet.
READ:WLAN:SUMMary:BURSt:POWer? (on page 1917)	Returns the average power of all symbols in the packet.
READ:WLAN:SUMMary:BURSt:POWer:CFACtor? (on page 1917)	Returns the peak-to-average burst power factor.
READ:WLAN:SUMMary:CERRor? (on page 1918)	Returns the symbol clock error.
READ:WLAN:SUMMary:CPE? (on page 1918)	Returns the RMS magnitude error of the pilots.
READ:WLAN:SUMMary:EVM:PEAK:ALL? (on page 1919)	Returns the peak EVM value for all subcarriers or all chips.
READ:WLAN:SUMMary:EVM:PEAK:CHIP:ALL? (on page 1919)	Returns the chip at which the reported peak EVM value occurred.
READ:WLAN:SUMMary:EVM:PEAK:CHIP:ONEKchips? (on page 1919)	Returns the chip at which the reported peak EVM value over the first 1000 chips occurred.
READ:WLAN:SUMMary:EVM:PEAK:DATA? (on page 1920)	Returns the peak EVM value for the data subcarriers.
READ:WLAN:SUMMary:EVM:PEAK:ONEKchips? (on page 1920)	Returns the peak EVM value measured over the first 1000 chips.
READ:WLAN:SUMMary:EVM:PEAK:PILots? (on page 1921)	Returns the peak EVM value for the pilots in the WLAN measurement.
READ:WLAN:SUMMary:EVM:PEAK:SCARrier:ALL? (on page 1921)	Returns the subcarrier on which the ALL peak EVM value occurred.
READ:WLAN:SUMMary:EVM:PEAK:SCARrier:DATA? (on page 1922)	Returns the subcarrier on which the DATA peak EVM value occurred.
READ:WLAN:SUMMary:EVM:PEAK:SCARrier:PILots? (on page 1922)	Returns the subcarrier on which the PILOT peak EVM value occurred.

Command	Description
READ:WLAN:SUMMary:EVM:PEAK:SYMBol:ALL? (on page 1922)	Returns the symbol on which the reported EVM value occurred.
READ:WLAN:SUMMary:EVM:PEAK:SYMBol:DATA? (on page 1923)	Returns the symbol on which the DATA peak EVM value occurred.
READ:WLAN:SUMMary:EVM:PEAK:SYMBol:PILots? (on page 1923)	Returns the symbol on which the PILOT peak EVM value occurred.
READ:WLAN:SUMMary:EVM:RMS:ALL? (on page 1924)	Returns the RMS EVM data for all subcarriers.
READ:WLAN:SUMMary:EVM:RMS:AVERage? (on page 1924)	Returns the average RMS EVM value.
READ:WLAN:SUMMary:EVM:RMS:DATA? (on page 1924)	Returns the RMS EVM data for the data subcarriers.
READ:WLAN:SUMMary:EVM:RMS:MAXimum? (on page 1925)	Returns the maximum RMS EVM value.
READ:WLAN:SUMMary:EVM:RMS:ONEKchips? (on page 1925)	Returns the RMS EVM data for the 1k chips in the WLAN measurement measured over the first 1000 chips.
READ:WLAN:SUMMary:EVM:RMS:PILots? (on page 1926)	Returns the RMS EVM data for the pilot subcarriers.
READ:WLAN:SUMMary:FERRor? (on page 1926)	Returns the difference between the measured carrier frequency of the signal and the measurement frequency setting.
READ:WLAN:SUMMary:HEADer? (on page 1927)	Returns the received packet header information.
READ:WLAN:SUMMary:HTSig? (on page 1927)	Returns the decimal values of the received packet HT-SIGNAL
READ:WLAN:SUMMary:IQOFfset? (on page 1927)	Returns the average magnitude of the DC subcarrier level relative to total signal power.
READ:WLAN:SUMMary:LSIG? (on page 1928)	Returns the decimal values of the received packet (legacy) SIGNAL symbols fields.
READ:WLAN:SUMMary:PACKet? (on page 1928)	Returns the type and number of symbols in the Preamble and Data portions of the packet.
READ:WLAN:SUMMary:PACKet:DMODulation? (on page 1928)	Returns the modulation used in the Data symbols.
READ:WLAN:SUMMary:PACKet:FORMat? (on page 1929)	Returns the packet format setting.
READ:WLAN:SUMMary:PACKet:GUARd? (on page 1929)	Returns the packet guard interval used by the DATA symbols.
READ:WLAN:SUMMary:SIG? (on page 1930)	Returns the decimal values of the received packet SIGNAL symbols fields.

Command	Description
READ:WLAN:SUMMary:VHTSig:A? (on page 1930)	Returns the decimal values of the received packet VHT-SIGNAL(A) symbols fields.
READ:WLAN:SUMMary:VHTSig:B? (on page 1931)	Returns the decimal values of the received packet VHT-SIGNAL(B) symbols fields.
READ:LTE subgroup	
READ:LTE:ACLR:MHITs:COUNt? (on page 1668)	Returns the number of rows in the results table.
READ:LTE:ACLR:MHITs<x>:BANDwidth? (on page 1668)	Returns the bandwidth value of the row (channel).
READ:LTE:ACLR:MHITs<x>:CHANnel:NAME? (on page 1669)	Returns the channel name of the specified index in the results table.
READ:LTE:ACLR:MHITs<x>:FREQuency:OFFSet? (on page 1669)	Returns the frequency offset of the channel.
READ:LTE:ACLR:MHITs<x>:INTeg:LOWEr:ABSolute? (on page 1670)	Returns the lower absolute value of the channel.
READ:LTE:ACLR:MHITs<x>:INTeg:LOWEr:RELative? (on page 1670)	Returns the lower relative value of the channel.
READ:LTE:ACLR:MHITs<x>:INTeg:UPPEr:ABSolute? (on page 1671)	Returns the upper absolute value of the channel.
READ:LTE:ACLR:REFerence:POWer? (on page 1671)	Returns the frequency and amplitude pairs of the spectrum trace.
READ:LTE:ACLR:RESUlts:STATus? (on page 1671)	Returns the pass or fail status.
READ:LTE:ACLR:SPECtrum:X? (on page 1672)	Returns the frequency of the spectrum trace.
READ:LTE:ACLR:SPECtrum:XY? (on page 1672)	Returns the frequency and amplitude pairs of the spectrum trace
READ:LTE:ACLR:SPECtrum:Y? (on page 1673)	Returns the amplitude of the spectrum trace.
READ:LTE:CHSPectrum:OBW? (on page 1673)	Returns the Occupied Bandwidth (Hz).
READ:LTE:CHSPectrum:POWer:CHANnel? (on page 1674)	Returns the Channel Power value.
READ:LTE:CHSPectrum:SPECtrum? (on page 1674)	Returns spectrum trace data.
READ:LTE:CONSte:CELL:ID? (on page 1675)	Queries the Cell ID value.
READ:LTE:CONSte:FREQuency:ERRor? (on page 1675)	Queries Frequency Error value in Hz.

Command	Description
READ:LTE:CONSte:GROUp:ID? (on page 1675)	Queries the Group ID value.
READ:LTE:CONSte:SECTor:ID? (on page 1676)	Queries the Sector ID value.
READ:LTE:PVTime:OFFSlot:POWer? (on page 1676)	Returns the TOff power measurement value in dBm/MHz.
READ:LTE:PVTime:RESUlts:STATus (on page 1677)	Returns the pass or fail status.
READ:LTE:PVTime:TRACe:X? (on page 1677)	Returns the horizontal values (time in seconds).
READ:LTE:PVTime:TRACe:XY? (on page 1677)	Returns the horizontal value (time in seconds) and vertical value (power).
READ:LTE:PVTime:TRACe:Y? (on page 1678)	Returns the vertical values (power).

Sense commands

Use the SENSe commands to set up detailed measurement conditions. See the Index for commands not listed in the following table.

The Record and Playback command subgroup may not apply to your instrument.Details can be found in the command descriptions.

Sense commands

Command	Description
[SENSe] basic command subgroup	**General analysis parameter control**
[SENSe]:ACQuisition:{BANDwidth|BWIDth} (on page 1940)	Sets or queries the acquisition bandwidth.
[SENSe]:ACQuisition:RECord command subgroup	**Record and Playback function for USB RSA products**
[SENSe]:ACQuisition:RECord:FILE:LENGth (on page 1947)	Sets or queries how much data (time in milliseconds, seconds, or minutes) is recorded when recording starts.
[SENSe]:ACQuisition:RECord:FILE:LENGth:ENABle (on page 1947)	Sets or queries the file length conditions to determine how much data is recorded.
[SENSe]:ACQuisition:RECord:FILE:LENGth:UNITs (on page 1948)	Sets or queries the units used for the file length setting.
[SENSe]:ACQuisition:RECord:FILE:LOCation (on page 1948)	Sets the file location of the saved data records.
[SENSe]:ACQuisition:RECord:FILE:MAXimum (on page 1949)	Sets or queries how many times a record is created, of the specified File Length setting.
[SENSe]:ACQuisition:RECord:FILE:NAME:BASE (on page 1949)	Sets or queries the base file name that is used for all data recordings.
[SENSe]:ACQuisition:RECord:FILE:STRucture (on page 1950)	Sets or queries the file structure to use for the recorded data files.

Command	Description
[SENSe]:ACQuisition:RECord:STOP (on page 1952)	Starts the data recording.
[SENSe]:ACQuisition:RECord:TYPE (on page 1952)	Sets or queries when the recording starts streaming data to the specified file.
[:SENSe]:ANALysis:BANDwidth:FILTer:MODE (on page 1956)	Sets or queries measurement bandwidth filter mode.
[SENSe]:AUDio:DEMod command subgroup	**Record and Playback function for USB RSA products**
[SENSe]:AUDio:DEMod:MUTE (on page 1964)	Sets or queries the mute setting for the audio demodulation.
[SENSe]:AUDio:DEMod[:STATe] (on page 1964)	Sets or queries the audio demodulation state.
[SENSe]:AUDio:DEMod:STReam:ACTivate (on page 1965)	Sets or queries the Stream to File function.
[SENSe]:AUDio:DEMod:STReam[:FILE] (on page 1965)	Sets the file name and file location of the audio file when streaming to disc.
[SENSe]:AUDio:DEMod:TYPE (on page 1966)	Sets or queries both the audio signal demodulation type and bandwidth.
[SENSe]:AUDio:DEMod:VOLume (on page 1966)	Sets or queries the volume of the demodulated audio signal when played out through the PC speakers.
[SENSe] TXGain command subgroup	**Transmission Gain measurements**
SENSe:TXGain:{BANDwidth|BWIDth}[:RESolution] (on page 2421)	Sets or queries the resolution bandwidth.
SENSe:TXGain:FREQuency:CENTer (on page 2422)	Sets or queries the frequency at the center of the selected Span.
SENSe:TXGain:FREQuency:POINts (on page 2422)	Sets or queries the number of frequency steps in the measurement range.
SENSe:TXGain:FREQuency:SPAN (on page 2423)	Sets or queries the difference between the start and stop frequencies (span).
SENSe:TXGain:FREQuency:STARt (on page 2423)	Sets or queries the lowest frequency in the span.
SENSe:TXGain:FREQuency:STEP (on page 2424)	Sets or queries the frequency increment.
SENSe:TXGain:FREQuency:STOP (on page 2424)	Sets or queries the highest frequency in the span.
[SENSe] Bluetooth command subgroup	**Bluetooth measurements**
[SENSe]:BLUEtooth:ANALysis:LENGth (on page 1987)	Sets or returns the analysis length.
[SENSe]:BLUEtooth:ANALysis:LENGth:ACTual? (on page 1987)	Sets the actual analysis length.
[SENSe]:BLUEtooth:ANALysis:LENGth:AUTO (on page 1988)	Sets or queries the length mode.

Command	Description
[SENSe]:BLUEtooth:ANALysis:MEAS[:BANDwidth| BWIDth] (on page 1988)	Sets or returns the measurement bandwidth.
[SENSe]:BLUEtooth:ANALysis:MEAS:SOURce (on page 1989)	Sets or returns the analysis measurement source.
[SENSe]:BLUEtooth:ANALysis:OFFSet (on page 1989)	Sets or returns the symbol analysis offset.
[SENSe]:BLUEtooth:ANALysis:OFFSet:AUTO (on page 1990)	Determines whether to set the carrier frequency error automatically or manually.
[SENSe]:BLUEtooth:ANALysis:TIMe:UNITs (on page 1990)	Sets the time units.
[SENSe]:BLUEtooth([:BANDwidth]|:BWIDth):MODE (on page 1990)	Sets or returns the measurement bandwidth mode.
[SENSe]:BLUEtooth([:BANDwidth]|:BWIDth): TINTerval (on page 1991)	Sets or returns the measurement bandwidth frequency span.
[SENSe]:BLUEtooth([:BANDwidth]|:BWIDth): TINTerval:AUTO (on page 1991)	Sets or returns the bandwidth to Auto or Manual.
[SENSe]:BLUEtooth:CONStellation:PREFs: GRATicule:SHOW (on page 1992)	Show or hides the graticule state.
[SENSe]:BLUEtooth:CONStellation:PREFs: MARKERS:SHOW (on page 1992)	Show or hides the marker readout.
[SENSe]:BLUEtooth:CONStellation:PREFs:RADIx (on page 1993)	Sets or returns the radix of the marker readout.
[SENSe]:BLUEtooth:CONStellation:TRACE<x>: CONTent (on page 1993)	Sets or returns the appearance of the specified trace (Trace 1 or Trace 2).
[SENSe]:BLUEtooth:CONStellation:TRACe<x>: FREEze (on page 1994)	Sets or freezes the updates of the specified trace (Trace 1 or Trace 2).
[SENSe]:BLUEtooth:CONSTellation:TRACE<x>: POINtsPerSymbol (on page 1994)	Sets or returns the number of points per symbol (how many points to use between symbols) for the specified trace (Trace 1 or Trace 2).
[SENSe]:BLUEtooth:CONStellation:TRACe<x>: SHOW (on page 1995)	Shows or hides the specified trace (Trace 1 or Trace 2).
[SENSe]:BLUEtooth:DELTa:AVERage:FONE (on page 1995)	Sets or returns the average frequency for the low deviation pattern.
[SENSe]:BLUEtooth:DELTa:AVERage:FTWO (on page 1995)	Sets or returns the average frequency for the high deviation pattern.
[SENSe]:BLUEtooth:FDVTime:TRACE:OCTEt? (on page 1996)	Returns the selected octet number on the Frequency Deviation vs. Time display.
[SENSe]:BLUEtooth:FDVTime:TRACE:OCTEt: TOTAL:COUNt? (on page 1996)	Returns the total octet value to be selected on the Frequency Deviation vs. Time display.

Command	Description
[SENSe]:BLUEtooth:FDVTime:TRACE:VIEW:MODE (on page 1997)	Sets or queries view mode in the Frequency Deviation vs. Time display.
[SENSe]:BLUEtooth:FILTer:ALPHa (on page 1997)	Sets or returns the alpha filter factor (α/BT).
[SENSe]:BLUEtooth:FILTer:MEASurement (on page 1998)	Sets or returns the measurement filter setting.
[SENSe]:BLUEtooth:FILTer:REFerence (on page 1998)	Sets or returns the reference filter setting.
[SENSe]:BLUEtooth:FREQuency:DEViation (on page 1999)	ets or returns the frequency deviation.
[SENSe]:BLUEtooth:FREQuency:DEViation:AUTO (on page 1999)	Determines whether to detect the frequency deviation automatically or manually.
[SENSe]:BLUEtooth:FREQuency:ERRor (on page 2000)	Sets or returns the frequency error in the Bluetooth analysis parameters (when Auto is off).
[SENSe]:BLUEtooth:FREQuency:ERRor:AUTO (on page 2000)	Determines whether the frequency error is set to Auto or Manual.
[SENSe]:BLUEtooth:INBEmissions:POWEr:LIMIt<x> (on page 2000)	Sets or queries the limits of the Bluetooth InBand Emission settings.
[SENSe]:BLUEtooth:POWer:CLASs (on page 2001)	Sets or queries the power class of the Bluetooth standard.
[SENSe]:BLUEtooth:STANdard (on page 2001)	Sets or queries the Bluetooth standard.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:DELTa:FONE:AVERage:MAX (on page 2002)	Sets or returns the maximum limit of the Basic Rate ΔF1 average.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:DELTa:FONE:AVERage:MIN (on page 2002)	Sets or returns the minimum limit of the Basic Rate ΔF1 average.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:DELTa:FTWO:PERcent (on page 2003)	Sets or returns the limit of the Basic Rate ΔF2 percentage.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:DELTa:FTWO:RESult (on page 2003)	Sets or returns the limit of the Basic Rate ΔF2 result.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:MAX:DRIFt:FNFZero (on page 2003)	Sets or returns the limit of the Basic Rate maximum drift fn–f0.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:MAX:FREQuency:OFFSet (on page 2004)	Sets or queries the limit of the Basic Rate maximum frequency offset.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:MAX:FREQuency:OFFSet:PREAmBDR (on page 2004)	Sets or queries the limit of the Basic Rate maximum frequency offset value from the preamble region.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:ONE:AVERage:MAX (on page 2005)	Sets or queries the limit of the Basic Rate Power Class 1 maximum average output power value.

Command	Description
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:ONE:AVERage:MIN (on page 2005)	Sets or queries the limit of the Basic Rate Power Class 1 minimum average output power value.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:ONE:PEAK:MAX (on page 2006)	Sets or queries the limit of the Basic Rate Power Class 1 maximum peak output power value.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:ONE:PEAK:MIN (on page 2006)	Sets or queries the limit of the Basic Rate Power Class 1 minimum peak output power value.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:THREe:AVERage:MAX (on page 2007)	Sets or queries the limit of the Basic Rate Power Class 3 maximum average output power value.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:THREe:AVERage:MIN (on page 2007)	Sets or queries the limit of the Basic Rate Power Class 3 minimum average output power value.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:THREe:PEAK:MAX (on page 2008)	Sets or queries the limit of the Basic Rate Power Class 3 maximum peak output power value.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:THREe:PEAK:MIN (on page 2008)	Sets or queries the limit of the Basic Rate Power Class 3 minimum peak output power value.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:TWO:AVERage:MAX (on page 2008)	Sets or queries the limit of the Basic Rate Power Class 2 maximum average output power value.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:TWO:AVERage:MIN (on page 2009)	Sets or queries the limit of the Basic Rate Power Class 2 minimum average output power value.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:TWO:PEAK:MAX (on page 2009)	Sets or queries the limit of the Basic Rate Power Class 2 maximum peak output power value.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:TWO:PEAK:MIN (on page 2010)	Sets or queries the limit of the Basic Rate Power Class 2 minimum peak output power value.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:DELTa:FONE:AVERage:MAX (on page 2014)	Sets or returns the maximum limit of the Low Energy ΔF1 average.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:DELTa:FONE:AVERage:MIN (on page 2010)	Sets or returns the minimum limit of the Low Energy ΔF1 average.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:DELTa:FTWO:PERcent (on page 2011)	Sets or returns the limit of the Low Energy ΔF2 percentage.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:DELTa:FTWO:RESult (on page 2011)	Sets or returns the limit of the Low Energy ΔF2 result.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:MAX:DRIFt:FNFZero (on page 2011)	Sets or returns the limit of the Low Energy maximum drift fn–f0.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:MAX:FREQuency:OFFSet (on page 2012)	Sets or returns the limit of the Low Energy maximum frequency offset.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:MAX:FREQuency:OFFSet:PREAmble (on page 2012)	Sets or returns the limit of the Low Energy maximum frequency offset value from the preamble region.

Command	Description
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:OPOWer:AVERage:MAX (on page 2013)	Sets or returns the limit of the Low Energy maximum average output power value.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:OPOWer:AVERage:MIN (on page 2013)	Sets or returns the limit of the Low Energy minimum average output power value.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:OPOWer:PEAK:MAX (on page 2014)	Sets or returns the limit of the Low Energy maximum peak output power value.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:OPOWer:PEAK:MIN (on page 2014)	Sets or returns the limit of the Low Energy minimum peak output power value.
[SENSe]:BLUEtooth:SUMMary:LIMIt:MAX:DRIFt: F1FZero (on page 2015)	Sets or returns the limit of the maximum fn–fn0 drift in the packet.
[SENSe]:BLUEtooth:SUMMary:LIMIt:MAX:DRIFt: FNFN5 (on page 2015)	Sets or returns the limit of the maximum fn–fn5 drift in the packet.
[SENSe]:BLUEtooth:SYMBol:POINts (on page 2017)	Sets or returns the number of points per symbol (how many points to use between symbols).
[SENSe]:BLUEtooth:TEST:PATTern (on page 2018)	Sets or returns the test pattern type in the Bluetooth displays.
[SENSe]:BLUEtooth:TEST:PATTern:AUTO:DETEct (on page 2018)	Sets or returns the auto detect setting for the Bluetooth displays.
[SENSe]:BLUEtooth:TIME:ZERO:REFERENCE (on page 2019)	Sets or returns the Time Zero Reference setting in the Bluetooth displays.
[SENSe]:BLUEtooth:TRACe:TYPE (on page 2019)	Sets or returns the trace type.
SENSe:EDIagram subgroup	**Eye diagram**
[SENSe]:EDIagram[:MEASview<y>]:SELEcted: SOURce (on page 2074)	Sets or queries the source of Eye Diagram for a selected Measurement view number.
SENSe:EVM subgroup	**EVM measurement**
[SENSe]:DEMOd:EVM[:MEASview<y>]:SELEcted: SOURce (on page 2075)	Sets or queries the source of EVM versus Time for a selected Measurement view number.
SENSe:FDVTime subgroup	**FDVTime measurement**
[SENSe]:FDVTime[:MEASview<y>]:SELEcted: SOURce (on page 2122)	Sets or queries the source of Frequency deviation versus Time for a selected Measurement view number
[SENSe]:MERRor subgroup	**MERRor measurement**
[SENSe]:MERRor[:MEASview<y>]:SELEcted: SOURce (on page 2191)	Sets or queries the source of Magnitude error versus Time for a selected Measurement view number
[SENSe]:NOISe subgroup	**Noise Figure and Gain measurements**

Command	Description
[SENSe]:NOISe:AVERage:COUNt (on page 2192)	Sets or queries the acquisition counts over which the noise average is computed.
[SENSe]:NOISe:BANDwidth (on page 2192)	Sets or queries the resolution bandwidth (RBW).
[SENSE]:NOISe:CONTrol:SETTling:OFF[:TIME] (on page 2193)	Sets or queries the analyzer off settle time.
[SENSE]:NOISe:CONTrol:SETTling:ON[:TIME] (on page 2193)	Sets or queries the analyzer on settle time.
[SENSe]:NOISe:CORRection:ENR:CONSTant (on page 2194)	Sets and queries the excess noise ratio (ENR) constant.
[SENSe]:NOISe:CORRection:ENR:MODE (on page 2194)	Sets and queries the Noise Source mode for the noise displays.
[SENSe]:NOISe:CORRection:ENR:TABLe:DATA? (on page 2195)	Queries the Noise Source model, serial number, and table values for frequency and ENR for the noise displays.
[SENSe]:NOISe:CORRection:ENR:TABLe:ID (on page 2195)	Sets or queries the Noise Source Model for the noise displays.
[SENSe]:NOISe:CORRection:ENR:TABLe:NEW (on page 2196)	Sets the Noise Source Model for the noise displays.
[SENSe]:NOISe:CORRection:ENR:TABLe:SERial (on page 2197)	Sets or queries the Noise Source serial number for the noise displays.
[SENSe]:NOISe:CORRection:TCOLd:ADUT: UREFerence (on page 2197)	Sets or queries the analyzer to use the reference temperature (K) for the loss at the DUT output in the noise displays.
[SENSe]:NOISe:CORRection:TCOLd:ADUT[:VALue] (on page 2198)	Sets or queries the temperature value (K) for the loss at the DUT output in the noise displays.
[SENSe]:NOISe:CORRection:TCOLd:BDUT: UREFerence (on page 2198)	Sets or queries the analyzer to use the reference temperature (K) for the loss at the DUT input in the noise displays.
[SENSe]:NOISe:CORRection:TCOLd:BDUT[:VALue] (on page 2199)	Sets or queries the temperature value (K) for the loss at the DUT input in the noise displays.
[SENSe]:NOISe:CORRection:TCOLd[:REFerence] (on page 2199)	Sets or queries the reference temperature value (K) in the display.
[SENSe]:NOISe:FIGure:POINTs:COUNt (on page 2200)	Sets the number of sample points for the display.
[SENSe]:NOISe:FREQuency:CENTer (on page 2200)	Sets or queries the center frequency value (Hz).
[SENSe]:NOISe:FREQuency:MODE (on page 2201)	Sets or queries the frequency mode in the display.
[SENSe]:NOISe:FREQuency:SPAN (on page 2201)	Sets or queries the frequency Span value (Hz).
[SENSe]:NOISe:FREQuency:STARt (on page 2202)	Sets or queries the Start Frequency value.

Command	Description
[SENSe]:NOISe:FREQuency:STEPs (on page 2202)	Sets or queries the frequency Steps value.
[SENSe]:NOISe:FREQuency:STOP (on page 2203)	Sets or queries the Stop Frequency value.
[SENSe]:NOISe:FREQuency:TABLe:DATA? (on page 2203)	Queries the data in the Frequency Table.
[SENSe]:NOISe:FREQuency:TABLe:NEW (on page 2204)	Creates a new Frequency Table with the specified values for the noise display.
[SENSe]:OBWidth:BOBW:XDBM:MEASurement: DIRection (on page 2208)	Sets or queries the measurement direction.
[SENSe]:OBWidth:BOBW:XDBM:RANGe (on page 2208)	Sets or queries the measurement range.
[SENSe]:P25 subgroup	**APCO P25 measurements**
[SENSe]:P25:ANALysis:OFFSet (on page 2224)	Sets or queries the offset value.
[SENSe]:P25:ANALysis:OFFSet:AUTO (on page 2225)	Sets or queries if the offset is set to Auto or if Auto is off.
[SENSe]:P25([:BANDwidth]|:BWIDth):MODE (on page 2225)	Sets or queries the Measurement BW (bandwidth) (frequency span) mode as Auto, Manual, or Link to Span.
[SENSe]:P25([:BANDwidth]|:BWIDth):TINTerval (on page 2226)	Sets or queries the Measurement BW (bandwidth) (frequency span).
[SENSe]:P25([:BANDwidth]|:BWIDth):TINTerval: AUTO (on page 2226)	Sets or queries the Measurement BW (bandwidth) (frequency span) mode as Auto or Manual.
[SENSe]:P25:FILTer:MEASurement (on page 2227)	Sets or queries the Measurement Filter setting.
[SENSe]:P25:FILTer:REFerence (on page 2228)	Sets or queries the Reference Filter setting.
[SENSe]:P25:FREQuency:ERRor (on page 2229)	Sets or queries the Frequency Offset (when Auto is off).
[SENSe]:P25:FREQuency:ERRor:AUTO (on page 2229)	Sets or queries if the Frequency Error is set to Auto or if Auto is off.
[SENSe]:P25:MODUlation:CHANnel:([BANDwidth]|:BWIDth)]? (on page 2230)	Queries the actual P25 Preset channel bandwidth.
[SENSe]:P25:MODUlation:STANdard (on page 2230)	Sets or queries the modulation standard.
SENSe:P25:MODUlation:TYPE (on page 2231)	Sets or queries the modulation type.
[SENSe]:P25:SRATe (on page 2231)	Sets or queries the symbol rate value for measurements.
[SENSe]:P25:SUMMary:CLEar:RESults (on page 2232)	Clears the results from the P25 Summary display.
[SENSe]:P25:SUMMary:LIMIt:FA (on page 2232)	Sets or queries the On/Off state of limit comparison and the upper limit value for Operating Frequency Accuracy.

Command	Description
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:AVERage:TERRor:ONE (on page 2233)	Sets or queries the On/Off state of limit comparison and the upper limit value for Average t_error_1.
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:AVERage:TERRor:ZERO (on page 2233)	Sets or queries the On/Off state of limit comparison and the upper limit value for Average t_error_0.
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:LOGIc:CHANnel:OFF:SLOT (on page 2234)	Sets or queries the On/Off state of limit comparison and the upper limit value for HCPM Transmitter Logic channel offslot.
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:LOGIc:CHANnel:PEAK:ACPR (on page 2234)	Sets or queries the On/Off state of limit comparison and the upper limit value for HCPM Transmitter Logic channel Peak ACPR.
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PEAK:NEGAtive:LO (on page 2235)	Sets or queries the On/Off state of limit comparison and the lower limit value for Frequency Deviation, negative peak.
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PEAK:NEGAtive:UP (on page 2236)	Sets or queries the On/Off state of limit comparison and the upper limit value for Frequency Deviation, negative peak.
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PEAK:POSItive:LO (on page 2236)	Sets or queries the On/Off state of limit comparison and the lower limit value for Frequency Deviation, positive peak.
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PEAK:POSItive:UP (on page 2237)	Sets or queries for the On/Off state of limit comparison and the upper limit value for Frequency Deviation, positive peak.
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PHASe[1/2]:F[1/2/3/4]:LO (on page 2238)	Sets or queries the On/Off state of limit comparison and the lower limit value for Frequency Deviation, f1/f2/f3/f4 for Phase1 or Phase2.
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PHASe[1/2]:F[1/2/3/4]:UP (on page 2238)	Sets or queries the On/Off state of limit comparison and the upper limit value for Frequency Deviation, f1/f2/f3/f4 for Phase1 or Phase2.
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMAX:OFF (on page 2239)	Sets or queries the On/Off state of limit comparison and the upper limit value for Pmax-off.
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMAX:ON (on page 2240)	Sets or queries the On/Off state of limit comparison and the upper limit value for Pmax-on.
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMAX:SS (on page 2241)	Sets or queries the On/Off state of limit comparison and the upper limit value for Pss-max.
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMIN:SS (on page 2242)	Sets or queries the On/Off state of limit comparison and the upper limit value for Pss-min.
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter:AVERage:ATTK:ENCOder (on page 2242)	Sets or queries the On/Off state of limit comparison and the upper limit value for Transmitter Average Attack encoder.
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter:AVERage:ATTK:ENCOder:BI (on page 2243)	Sets or queries the On/Off state of limit comparison and the upper limit value for Transmitter Average Attack encoder, busy/idle.
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter: AVERage:ATTK:POWEr (on page 2243)	Sets or queries the On/Off state of limit comparison and the upper limit value for Transmitter Average Attack power.

Command	Description
[SENSe]:P25:SUMMary:LIMIt:FREQuency: DEVIation:TRANsmitter: AVERage:ATTK:POWEr:BI (on page 2244)	Sets or queries for the On/Off state of limit comparison and the upper limit value for Transmitter Average Attack power, busy/idle.
[SENSe]:P25:SUMMary:LIMIt:FREQuency: DEVIation:TRANsmitter: AVERage:THRUput:DLAY (on page 2245)	Sets or queries the On/Off state of limit comparison and the upper limit value for Transmitter Average throughput delay.
[SENSe]:P25:SUMMary:LIMIt:MF (on page 2245)	Sets or queries the On/Off state of limit comparison and the upper limit value for Modulation Fidelity.
[SENSe]:P25:SUMMary:LIMIt:RF:OUTPower (on page 2246)	Sets or queries the On/Off state of limit comparison and the upper limit value for RF Output Power.
[SENSe]:P25:SUMMary:LIMIt:SRA (on page 2246)	Sets or queries the On/Off state of limit comparison and the upper limit value for Symbol Rate Accuracy.
[SENSe]:P25:SYMBols:POINts (on page 2247)	Sets or queries the Points/Symbols to display.
[SENSe]:P25:TIMe:UNITs (on page 2248)	Sets or queries the fundamental unit of time for the P25 measurements.
[SENSe]:ACQuisition subgroup	**Acquisition measurements**
[SENSe]:SIGNalvu:ACQuisition:FFRame:ACTual? (on page 2413)	Queries the actual number of Fast Frames.
[SENSe]:SIGNalvu:ACQuisition:FFRame:LIMit (on page 2414)	Sets or queries the limit number of Fast Frames.
[SENSe]:SIGNalvu:ACQuisition:FFRame:STATe (on page 2414)	Determines whether to enable or disable the Fast Frame.
[SENSe]:ACQuisition:FSAVe:ENAble (on page 1942)	Set or queries the FastSave acquisition mode.
[SENSe]:ACQuisition:FSAVe:FILEs:MAXimum (on page 1942)	Set or queries the FastSave maximum number of files saved per run.
[SENSe]:ACQuisition:FSAVe:FORMat (on page 1943)	Set or queries the FastSave file format.
[SENSe]:ACQuisition:FSAVe:LOCation (on page 1943)	Set or queries the FastSave file save location (a directory).
[SENSe]:ACQuisition:FSAVe:NAME:BASE (on page 1944)	Set or queries the FastSave Base file name.
[SENSe]:ACQuisition:MEMory:AVAilable:SAMPles? (on page 1944)	Returns the amount of acquisition memory available in the instrument.
[SENSe]:ACQuisition:MEMory:CAPacity[:TIME]? (on page 1944)	Returns the acquisition memory capacity.
[SENSe]:ACQuisition:MEMory:USED[:PERCent]? (on page 1945)	Returns the percentage of the capacity used.

Command	Description
[SENSe]:ACQuisition:MODE (on page 1945)	Sets or queries the acquisition mode.
[SENSe]:ACQuisition:SAMPles (on page 1952)	Sets or queries the acquisition samples.
[SENSe]:ACQuisition:SEConds (on page 1953)	Sets or queries the acquisition length.
[SENSe]:ANALysis:ADVanced:DITHer (on page 1955)	Determines whether to enable or disable dithering.
[SENSe]:ANALysis:ADVanced:DITHer:HWARe: STATus? (on page 1956)	Returns the dithering hardware status.
[SENSe]:ANALysis:LENGth[:SOURce<z>]:ACTual (on page 1957)	Sets or queries the analysis length. Programming a specified length sets [SENSe]:ANALysis:AUTO OFF.
[SENSe]:ANALysis:LENGth:ACTual?	Queries the actual analysis length.
[SENSe]:ANALysis:LENGth[:SOURce<z>]:AUTO (on page 1958)	Determines whether to set the analysis length automatically or manually.
[SENSe]:ANALysis:REFerence[:SOURce<z>] (on page 1958)	Sets or queries the analysis time reference.
[SENSe]:ANALysis:STARt[:SOURce<z>] (on page 1959)	Sets or queries the analysis offset time. Programming a specified offset time sets [SENSe]:ANALysis:STARt:AUTO OFF.
[SENSe]:ANALysis:STARt[:SOURce<z>]:AUTO (on page 1959)	Determines whether to set the analysis offset automatically or manually.
[SENSe]:ANSPectrum:STARt:AUTO:METHod (on page 1960)	Sets or queries the method used for computing auto analysis and spectrum offsets.
[SENSe]:MEASurement:FREQuency[:SOURce<z>] (on page 2190)	Sets or queries the measurement frequency.
[SENSe]:MEASurement:FREQuency[:SOURce<z>]:CENTer:LOCK (on page 2191)	Sets or queries locking the measurement center frequency.
[SENSe]:POWer:UNITs (on page 2269)	Sets or queries the fundamental unit of power.
[SENSe]:ROSCillator:SOURce (on page 2323)	Sets or queries the reference oscillator source.
[SENSe]:SIGNalpath:PRIority (on page 2364)	Sets or queries the signal path priority for the measurements (for RSA5000 series only).
[SENSe]:USETtings (on page 2425)	Updates the analyzer settings.
[SENSe]:ACPower subgroup	**Channel power and ACPR measurement**
[SENSe]:ACPower:AVERage (on page 1931)	Sets or queries how to average waveform.
[SENSe]:ACPower:AVERage:COUNt (on page 1932)	Sets or queries the number of traces for averaging.

Command	Description
[SENSe]:ACPower:{BANDwidth|BWIDth}[: RESolution] (on page 1932)	Sets or queries the resolution bandwidth (RBW).
[SENSe]:ACPower:{BANDwidth|BWIDth}[: RESolution]:ACTual? (on page 1932)	Queries the actual resolution bandwidth (RBW).
[SENSe]:ACPower:{BANDwidth|BWIDth}[: RESolution]:AUTO (on page 1933)	Sets or queries whether to set the RBW automatically.
[SENSe]:ACPower:{BANDwidth|BWIDth}:VIDeo (on page 1933)	Sets or queries the video bandwidth (VBW).
[SENSe]:ACPower:{BANDwidth|BWIDth}:VIDeo: STATe (on page 1934)	Sets or queries whether to enable the video bandwidth (VBW).
[SENSe]:ACPower:CHANnel:{BANDwidth|BWIDth} (on page 1934)	Sets or queries the channel bandwidth.
[SENSe]:ACPower:CHANnel:FILTer (on page 1935)	Sets or queries the adjacent channel filter.
[SENSe]:ACPower:CHANnel:PAIRs (on page 1935)	Sets or queries the number of adjacent channel pairs.
[SENSe]:ACPower:CHANnel:SPACing (on page 1936)	Sets or queries the channel-to-channel spacing.
[SENSe]:ACPower:CHIPrate (on page 1936)	Sets or queries the chip rate.
[SENSe]:ACPower:CLEar:RESults (on page 1936)	Restarts the average trace.
[SENSe]:ACPower:FREQuency (on page 1937)	Sets or queries the center frequency.
[SENSe]:ACPower:FREQuency:STEP (on page 1937)	Sets or queries the frequency step size.
[SENSe]:ACPower:FREQuency:STEP:AUTO (on page 1938)	Sets or queries whether to set the frequency step size automatically.
[SENSe]:ACPower:NFLoor:STATe (on page 1938)	Sets or queries whether to enable the correction for noise floor.
[SENSe]:ACPower:POINts:COUNt (on page 1939)	Sets or queries the number of trace points.
[SENSe]:ACPower:RRCRolloff (on page 1939)	Sets or queries the filter parameter for the Root Raised Cosine filter.
[SENSe]:ACPower:SYMBrate (on page 1940)	Sets or queries the symbol rate.
[SENSe]:{AM|FM|PM} subgroup (Option 21 only)	**AM/FM/PM measurement**
[SENSe]:{AM|FM|PM}:{BANDwidth|BWIDth}: MEASurement (on page 1953)	Sets measurement bandwidth for the AM, FM, or PM demodulation to analyze.
[SENSe]:{AM|FM|PM}:CLEar:RESults (on page 1954)	Clears measurement results.
[SENSe]:{AM|FM|PM}:{MTPoints|MAXTracepoints} (on page 1954)	Sets or queries the maximum trace points.

Command	Description
[SENSe]:AM:DETect:AMPLitude (on page 1955)	Sets or queries the carrier amplitude detection method.
[SENSe]:{FM|PM}:BURSt:THReshold (on page 2122)	Sets or queries the threshold level to detect bursts.
[SENSe]:{FM|PM}:FREQuency:OFFSet (on page 2123)	Sets or queries the carrier frequency offset in the FM/PM measurement.
[SENSe]:{FM|PM}:FREQuency:OFFSet:MARKer (on page 2123)	Sets the frequency offset from selected marker or delta marker location.
[SENSe]:{FM|PM}:FREQuency:SEARch:AUTO (on page 2124)	Sets or queries whether to detect the carrier frequency automatically.
[SENSe]:PM:PHASe:OFFSet (on page 2264)	Sets the phase offset from the selected marker location.
[SENSe]:PM:PHASe:OFFSet:MARKer (on page 2264)	Sets the carrier frequency offset from the selected marker.
[SENSe]:PM:PHASe:SEARch:AUTO (on page 2265)	Sets or queries whether to detect the phase automatically or manually.
{SENSe]:AUDio subgroup	**Audio measurement**
[SENSe]:AUDio:AVERage:COUNt (on page 1960)	Sets or queries the number of harmonics to average in the audio measurement.
[SENSe]:AUDio:AVERage:ENABle (on page 1961)	Enables or disables averaging of harmonics in the audio measurement.
[SENSe]:AUDio:BANDwidth|BWIDth (on page 1961)	Sets or queries the audio measurement bandwidth.
[SENSe]:AUDio:BANDwidth|BWIDth:RESolution (on page 1962)	Sets or queries the resolution bandwidth for the audio measurement.
[SENSe]:AUDio:BANDwidth|BWIDth:RESolution: AUTO (on page 1962)	Sets or queries the resolution bandwidth for the audio measurement.
[SENSe]:AUDio:BANDwidth|BWIDth:RESolution: SHAPe (on page 1962)	Sets or queries the shape of the resolution bandwidth for the audio measurement.
[SENSe]:AUDio:CARRier:OFFSet (on page 1963)	Sets or queries the carrier frequency offset.
[SENSe]:AUDio:CARRier:OFFSet:AUTO (on page 1963)	Determines whether to set the carrier frequency error automatically or manually.
[SENSe]:AUDio:FILTer:DEEMphasis (on page 1967)	Sets or queries the de-emphasis filter type.
[SENSe]:AUDio:FILTer:DEEMphasis:ENABle (on page 1967)	Enables or disables the de-emphasis filter for the audio measurement.
[SENSe]:AUDio:FILTer:DEEMphasis:USER (on page 1968)	Sets or queries a user-defined de-emphasis audio filter.
[SENSe]:AUDio:FILTer:HPF (on page 1968)	Sets or queries the high-pass filter type.

Command	Description
[SENSe]:AUDio:FILTer:HPF:ENABle (on page 1969)	Enables or disables the high-pass audio filter for the audio measurement.
[SENSe]:AUDio:FILTer:HPF:USER (on page 1969)	Sets or queries a user-defined high-pass audio filter.
[SENSe]:AUDio:FILTer:DEEMphasis:USER (on page 1968)	Sets or queries a user-defined high-pass audio filter.
[SENSe]:AUDio:FILTer:LPF (on page 1969)	Sets or queries the low-pass filter type.
[SENSe]:AUDio:FILTer:LPF:ENABle (on page 1970)	Enables or disables the low-pass filter for the audio measurement.
[SENSe]:AUDio:FILTer:LPF:USER (on page 1970)	Sets or queries a user-defined low-pass audio filter.
[SENSe]:AUDio:FILTer:MODE:PREDefined (on page 1971)	Selects a predefined audio filter or a filter stored on the instrument file.
[SENSe]:AUDio:FILTer:SOURce? (on page 1971)	Queries the measurement filter from the user-defined source file.
[SENSe]:AUDio:FILTer:STANdard (on page 1972)	Sets or queries the Standard audio filter.
[SENSe]:AUDio:FILTer:STANdard:ENABle (on page 1972)	Enables or disables a Standard filter for the audio measurement.
[SENSe]:AUDio:FREQuency (on page 1973)	Sets or queries the reference audio frequency.
[SENSe]:AUDio:FREQuency:AUTO (on page 1973)	Determines whether to set the reference audio frequency automatically or manually.
[SENSe]:AUDio:HNOise:ENABle (on page 1973)	Enables or disables the Hum and Noise information in the Audio Summary display.
[SENSe]:AUDio:HNOise:REFerence (on page 1974)	Captures the current Hum and Noise information in the Audio Summary display.
[SENSe]:AUDio:SIGnal:TYPE (on page 1974)	Sets or queries the audio signal type.
[SENSe]:AUDio:SPECtrum:CLEar:RESults (on page 1975)	Clears the results of the audio spectrum measurement.
[SENSe]:AUDio:SPECtrum:POINts:COUNt (on page 1975)	Sets or queries the number of trace points acquired for the audio spectrum display.
[SENSe]:AUDio:UNITs (on page 1976)	Sets or queries the units of the audio spectrum measurement.
[SENSe]:AVTime subgroup	**Amplitude versus Time measurement**
[SENSe]:AVTime[:MEASview<y>]:{BANDwidth| BWIDth} (on page 1978)	Sets or queries the time-domain bandwidth filter.
[SENSe]:AVTime[:MEASview<y>]:{BANDwidth| BWIDth}:ACTual? (on page 1978)	Queries the actual resolution bandwidth (RBW).
[SENSe]:AVTime[:MEASview<y>]:CLEar:RESults (on page 1979)	Restarts multi-trace functions (Average and Max/Min Hold).

Command	Description
[SENSe]:AVTime[:MEASview<y>]:COLor (on page 1979)	Sets or queries the color palette of three-dimensional graphs.
[SENSe]:AVTime[:MEASview<y>]:COLor:MAXimum (on page 1980)	Sets or queries the maximum value of the color axis in the Amplitude versus Time measurement.
[SENSe]:AVTime[:MEASview<y>]:COLor:MINimum (on page 1981)	Sets or queries the minimum value of the color axis in the Amplitude versus Time measurement.
[SENSe]:AVTime[:MEASview<y>]:COLor:REVerse[:STATe] (on page 1981)	Determines whether to reverse the color palette in the Amplitude versus Time view. When reversed, the highest density takes the color normally used for the lowest.
[SENSe]:AVTime[:MEASview<y>]:MAXTracepoints (on page 1982)	Sets or queries the maximum trace points.
[SENSe]:AVTime[:MEASview<y>]:METHod (on page 1982)	Sets or queries the method to set the measurement bandwidth.
[SENSe]:AVTime[:MEASview<y>]:OGRAm:COLor (on page 1983)	Sets or queries the ogram color scheme.
[SENSe]:AVTime[:MEASview<y>]:OGRAm:COLor:MAXimum (on page 1984)	Sets or queries the maximum value of the color axis in the ogram.
[SENSe]:AVTime[:MEASview<y>]:OGRAm:COLor:MINimum (on page 1984)	Sets or queries the minimum value of the color axis in the ogram.
[SENSe]:AVTime[:MEASview<y>]:OGRAm:COLor:REVerse (on page 1985)	Sets or queries reverse color scheme state.
[SENSe]:AVTime[:MEASview<y>]:PLOT (on page 1986)	Sets or queries the Amplitude vs time display mode.
[SENSe]:AVTime[:MEASview<y>]:SPAN (on page 1986)	Sets or queries the frequency span.
[SENSe]:CCDF subgroup	**CCDF measurement**
[SENSe]:CCDF:{BANDwidth|BWIDth} (on page 2020)	Sets or queries the CCDF measurement bandwidth.
[SENSe]:CCDF:CLEar (on page 2020)	Clears the CCDF accumulator and restarts the measurement.
[SENSe]:CCDF:TIME:TOTal:LENGth (on page 2021)	Sets or queries the CCDF measurement time.
[SENSe]:CCDF:TIME:TYPE (on page 2021)	Sets or queries how to repeat the CCDF measurement.
[SENSe]:CONSte[:MEASview<y>]:SELEcted: SOURce (on page 2020)	Sets or queries the Source of Constellation for a selected Measurement view number.
[SENSe]:DDEMod subgroup (Option 21 only)	**General purpose digital modulation measurements**
[SENSe]:DDEMod[:SOURce<x>]:ANALysis:LENGth (on page 2022)	Sets or queries the analysis length.
[SENSe]:DDEMod[:SOURce<x>]:ANALysis:LENGth:ACTual? (on page 2023)	Queries the actual analysis length.

Command	Description
[SENSe]:DDEMod[:SOURce<x>]:ANALysis:LENGth: AUTO (on page 2023)	Sets or queries whether to set the analysis length automatically.
[SENSe]:DDEMod[:SOURce<x>]:{BANDwidth|BWIDth}:MODE (on page 2024)	Sets or queries the measurement bandwidth mode.
[SENSe]:DDEMod[:SOURce<x>]:{BANDwidth|BWIDth}:TINTerval (on page 2024)	Sets or queries the measurement bandwidth (frequency span).
[SENSe]:DDEMod[:SOURce<x>]:{BANDwidth|BWIDth}:TINTerval:AUTO (on page 2025)	Sets the measurement bandwidth (frequency span) automatically.
[SENSe]:DDEMod[:SOURce<x>]:BURSt:DETect (on page 2026)	Sets or queries how to detect bursts.
[SENSe]:DDEMod[:SOURce<x>]:BURSt:THReshold (on page 2026)	Sets or queries the threshold level to determine a burst.
[SENSe]:DDEMod[:SOURce<x>]:CARRier:OFFSet (on page 2027)	Sets or queries the carrier frequency offset.
[SENSe]:DDEMod[:SOURce<x>]:CARRier:OFFSet: AUTO (on page 2028)	Sets or queries whether to detect the carrier frequency automatically.
[SENSe]:DDEMod[:SOURce<x>]:EQUalizer: AVAilable? (on page 2028)	Determines if the equalizer is available.
[SENSe]:DDEMod[:SOURce<x>]:EQUalizer: CONVergence (on page 2029)	Sets or queries the Convergence value (also known as tap update rate).
[SENSe]:DDEMod[:SOURce<x>]:EQUalizer:ENABle (on page 2029)	Sets or queries the Equalizer state.
[SENSe]:DDEMod[:SOURce<x>]:EQUalizer:LENGth (on page 2030)	Sets or queries the equalizer filter length.
[SENSe]:DDEMod[:SOURce<x>]:EQUalizer:MODE (on page 2030)	Sets or queries the equalizer mode. The equalizer can in learning mode (train) or enabled (hold).
[SENSe]:DDEMod[:SOURce<x>]:EQUalizer:RESet (on page 2031)	Resets the equalizer settings to default values.
[SENSe]:DDEMod[:SOURce<x>]:EQUalizer:TAPS (on page 2031)	Sets or queries the number of filter coefficients for the equalizer.
[SENSe]:DDEMod[:SOURce<x>]:EQUalizer:TSRAtio (on page 2031)	Sets or queries the Equalizer Taps/Symbol parameter.
[SENSe]:DDEMod[:SOURce<x>]:FILTer:ALPHa (on page 2032)	Sets or queries the filter factor (a/BT).
[SENSe]:DDEMod[:SOURce<x>]:FILTer: MEASurement (on page 2032)	Sets or queries the measurement filter.

Command	Description
[SENSe]:DDEMod[:SOURce<x>]:FILTer:REFerence (on page 2033)	Sets or queries the reference filter.
[SENSe]:DDEMod[:SOURce<x>]:FREQuency: DEViation (on page 2034)	Sets or queries the frequency deviation to determine two states for FSK.
[SENSe]:DDEMod[:SOURce<x>]:FREQuency: DEViation:AUTO (on page 2035)	Sets or queries whether to detect the frequency deviation automatically.
[SENSe]:DDEMod[:SOURce<x>]:MAGNitude: NORMalize (on page 2035)	Sets or queries the method for the magnitude normalization.
[SENSe]:DDEMod[:SOURce<x>]:MINDex (on page 2036)	Sets or queries the modulation index for a CPM signal.
[SENSe]:DDEMod[:SOURce<x>]:MINDex:AUTO (on page 2037)	Sets or queries whether to detect the modulation index automatically.
[SENSe]:DDEMod[:SOURce<x>]:MODulation:TYPE (on page 2037)	Sets or queries the modulation type.
[SENSe]:DDEMod[:SOURce<x>]:PRESet (on page 2039)	Presets the modulation analysis to a communication standard.
[SENSe]:DDEMod[:SOURce<x>]:RING:RADius (on page 2042)	Sets or queries the APSK radius parameter.
[SENSe]:DDEMod[:SOURce<x>]:RING:ROTation (on page 2042)	Sets or queries the APSK rotation parameter.
[SENSe]:DDEMod[:SOURce<x>]:SRATe (on page 2043)	Sets or queries the symbol rate.
[SENSe]:DDEMod[:SOURce<x>]:SWAP:IQ (on page 2043)	Sets or queries whether or not to swap I and Q data.
[SENSe]:DDEMod[:SOURce<x>]:SYMBol:HSSHift (on page 2044)	Sets or queries the Q data half-symbol shift for OQPSK and SOQPSK signals.
[SENSe]:DDEMod[:SOURce<x>]:SYMBol:MAP: SOURce? (on page 2044)	Specifies or queries the user symbol map file.
[SENSe]:DDEMod[:SOURce<x>]:SYMBol:MAP[: STATe] (on page 2045)	Sets or queries whether to use the user symbol map.
[SENSe]:DDEMod[:SOURce<x>]:SYMBol:PLOT: POSition (on page 2045)	Sets or queries the symbol point location on an SOQPSK waveform.
[SENSe]:DDEMod[:SOURce<x>]:SYMBol:POINts (on page 2046)	Sets or queries the number of points per symbol.
[SENSe]:DDEMod[:SOURce<x>]:SYMBol:RATE: SEARch (on page 2047)	Determines whether to enable a symbol rate search.
[SENSe]:DDEMod[:SOURce<x>]:SYNCh:WORD (on page 2047)	Sets or queries whether to enable the synchronization word.

Command	Description
[SENSe]:DDEMod[:SOURce<x>]:SYNCh:WORD: SYMBol (on page 2048)	Sets or queries the synchronization word.
[SENSe]:DDEMod[:SOURce<x>]:TIMe:UNITs (on page 2048)	Sets or queries the fundamental unit of time.
[SENSe]:DIQVtime subgroup	**Demodulated I&Q versus time measurement view number**
[SENSe]:DIQVtime[:MEASview<y>]:SELEcted: SOURce (on page 2049)	Sets or queries the source for a selected Measurement view number.
[SENSe]:DPX subgroup	**DPX spectrum measurement**
[SENSe]:DPX:AUDio:DEMod:GAIN (on page 2049)	Sets or queries the gain for the audio demodulation.
[SENSe]:DPX:AUDio:DEMod:RXBWidth (on page 2050)	Sets or queries the receiver bandwidth for the audio demodulation.
[SENSe]:DPX:AUDio:DEMod:RXFRequency? (on page 2050)	Queries the receiver frequency for the audio demodulation.
[SENSe]:DPX:AUDio:DEMod:STATe (on page 2051)	Sets or queries whether to enable or disable the audio demodulation.
[SENSe]:DPX:AUDio:DEMod:TUNE (on page 2051)	Sets or queries how to determine the tuning frequency.
[SENSe]:DPX:AUDio:DEMod:TYPE (on page 2052)	Sets or queries the modulation type for the audio demodulation.
[SENSe]:DPX:{BANDwidth|BWIDth}:ACTual? (on page 2052)	Sets or queries the resolution bandwidth (RBW).
[SENSe]:DPX:{BANDwidth|BWIDth}:RESolution (on page 2053)	Sets or queries the resolution bandwidth.
[SENSe]:DPX:{BANDwidth|BWIDth}[:RESolution]: AUTO (on page 2053)	Determines whether to set the resolution bandwidth (RBW) automatically or manually in the DPX spectrum measurement.
[SENSe]:DPX:FREQuency:SPAN: {BANDwidth|BWIDth}[:RESolution]:RATio (on page 2064)	Sets or queries whether to set the RBW automatically.
[SENSe]:DPX:CLEar:RESults (on page 2053)	Restarts multi-trace functions (Average and Max/Min Hold).
[SENSe]:DPX:COLor (on page 2054)	Sets or queries the color palette of three-dimensional graphs.
[SENSe]:DPX:COLor:MAXimum (on page 2055)	Sets or queries the maximum value of the color axis.
[SENSe]:DPX:COLor:MINimum (on page 2055)	Sets or queries the minimum value of the color axis.
[SENSe]:DPX:COLor:REVerse[:STATe] (on page 2056)	Determines whether to reverse the color palette in the DPX, Spectrum view.
[SENSe]:DPX:DDENsity:MEASurement:SHOW (on page 2056)	Sets or queries the Show measurement value (measurement always visible for DPX Trigger).
SENSe:DPX:DGRam:COLor (on page 2056)	Sets or queries the color palette of three-dimensional graphs.

Command	Description
[SENSe]:DPX:DGRam:COLor:MAXimum (on page 2057)	Sets or queries the maximum value of the color axis in the DPXogram measurement.
[SENSe]:DPX:DGRam:COLor:REVerse (on page 2058)	Sets or queries reverse color scheme state.
[SENSe]:DPX:DGRam:COLor:MINimum (on page 2058)	Sets or queries the minimum value of the color axis in the DPXogram measurement.
[SENSe]:DPX:DGRam:POINts:COUNt (on page 2059)	Sets the number of trace points acquired for the DPXogram display.
[SENSe]:DPX:DGRam:TIME:CAPacity? (on page 2059)	Returns the time capacity of the DPXogram measurement.
[SENSe]:DPX:DGRam:TIME:RESolution (on page 2060)	Sets the time resolution of the DPXogram measurement.
[SENSe]:DPX:DGRam:TIME:RESolution:AUTO (on page 2060)	Sets the time resolution automatically or manually in the DPXogram measurement.
[SENSe]:DPX:DGRam:WATerfall:DIRection (on page 2061)	Sets or queries the direction of the Waterfall (DPXogram) display.
[SENSe]:DPX:DGRam:WATerfall:ENABle (on page 2061)	Sets or queries the Waterfall (DPXogram) display.
[SENSe]:DPX:DGRam:Y[:SCALe]:AUTO (on page 2061)	Rescales the height axis automatically to fit the waveform to the screen in the DPXogram display.
[SENSe]:DPX:DGRam:Y[:SCALe]:RESet (on page 2062)	Resets the height scale of the DPXogram display.
[SENSe]:DPX:DWELl (on page 2062)	Sets or queries the value of the Dwell time for the DPX spectrum measurement.
[SENSe]:DPX:DWELl:AUTO (on page 2063)	Sets the value of the Dwell time automatically.
[SENSe]:DPX:PLOT (on page 2067)	Sets or queries the plot in the DPX spectrum measurement.
[SENSe]:DPX:POINts:COUNt (on page 2067)	Sets or queries the number of DPX trace points.
[SENSe]:DPX:FREQuency:CENTer (on page 2063)	Sets or queries the center frequency.
[SENSe]:DPX:FREQuency:SPAN (on page 2064)	Sets or queries the frequency span.
[SENSe]:DPX:FREQuency:STARt (on page 2065)	Sets or queries the measurement start frequency.
[SENSe]:DPX:FREQuency:STEP (on page 2065)	Sets or queries the frequency step size.
[SENSe]:DPX:FREQuency:STEP:AUTO (on page 2066)	Sets or queries whether to set the frequency step size automatically.
[SENSe]:DPX:FREQuency:STOP (on page 2066)	Sets or queries the measurement stop frequency.
[SENSe]:DPX:TDM:{BANDwidth|BWIDth} (on page 2068)	Sets or queries the value of the measurement bandwidth in the DPX measurement.

Command	Description
[SENSe]:DPX:TDM:{BANDwidth|BWIDth}:ACTual? (on page 2068)	Queries the actual bandwidth in the DPX measurement.
[SENSe]:DPX:TDM:FREQuency:CENTer (on page 2069)	Sets or queries the center frequency in the DPX spectrum measurement.
[SENSe]:DPX:TDM:FREQuency:TXBer:CLEar: RESults (on page 2069)	Clears the measurement results of the active Tx BER measurement.
[SENSe]:DPX:TDM:FREQuency:TXBer:ENABle (on page 2070)	Sets or queries the Tx BER measurement.
[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern (on page 2070)	Sets or queries the Tx BER data pattern.
[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern: INVert (on page 2071)	Sets or queries the data pattern Invert setting.
[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern: REPeat (on page 2071)	Sets or queries the test duration for a User Pattern.
[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern: REVerse (on page 2072)	Sets or queries whether the data pattern is in normal order or reversed.
[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern: SOURce? (on page 2072)	Returns the path and file name of the user-defined pattern file.
[SENSe]:DPX:TDM:RBW (on page 2073)	Sets or queries the value of the RBW filter.
[SENSe]:DPX:TDM:RBW:STATe (on page 2073)	Determines whether to set the resolution bandwidth (RBW) automatically or manually.
[SENSe]:DPX:TDM:SWEep:TIME (on page 2074)	Sets or queries the sweep time.
[SENSe]:DPX:TDM:TMOTion (on page 2074)	Sets or queries the scroll direction.
[SENSe]:{FSETtling|PSETtling} subgroup	**Settling time measurements**
[SENSe]:{FSETtling|PSETtling}:{BANDwidth| BWIDTH} (on page 2124)	Sets or queries the measurement bandwidth for the Settling Time measurements.
[SENSe]:{FSETtling|PSETtling}:{BANDwidth| BWIDTH}:ACTual? (on page 2125)	Queries the actual measurement bandwidth in the Settling Time measurements.
[SENSe]:{FSETtling|PSETtling}:FREQuency:CENTer (on page 2125)	Queries the actual measurement bandwidth in the Settling Time measurements.
[SENSe]:{FSETtling|PSETtling}:FREQuency:OFFSet (on page 2126)	Sets or queries the offset frequency.
[SENSe]:{FSETtling|PSETtling}:LENGth (on page 2126)	Sets or queries the measurement length.

Command	Description
[SENSe]:{FSETtling|PSETtling}:LENGth:ACTual? (on page 2126)	Queries the actual measurement length for the Settling Time measurements.
[SENSe]:{FSETtling|PSETtling}:MASK:LIMit (on page 2127)	Sets or queries the frequency or phase limit for the specified region.
[SENSe]:{FSETtling|PSETtling}:MASK:STATe (on page 2128)	Sets or queries mask test state.
[SENSe]:{FSETtling|PSETtling}:MASK:TIME: REFerence (on page 2128)	Sets or queries the mask time reference.
[SENSe]:{FSETtling|PSETtling}:MASK:TIME:STARt (on page 2128)	Sets or queries the mask region start time in seconds.
[SENSe]:{FSETtling|PSETtling}:MASK:TIME:STOP (on page 2129)	Sets or queries the mask region stop time in seconds.
[SENSe]:{FSETtling|PSETtling}:MAXTracepoints (on page 2129)	Sets or queries the maximum trace points in the Settling Time measurements.
[SENSe]:{FSETtling|PSETtling}:SDURation:MINimum (on page 2130)	Sets or queries the minimum settled duration in seconds.
[SENSe]:{FSETtling|PSETtling}:TARGet:REFerence (on page 2130)	Sets or queries the target reference frequency.
[SENSe]:{FSETtling|PSETtling}:TOLerance (on page 2131)	Sets or queries the frequency or phase range into which the signal must settle.
[SENSe]:FVTime subgroup	**Frequency versus Time measurement**
[SENSe]:FVTime[:MEASview<y>]:CLEar:RESults (on page 2131)	Restarts multi-trace functions (Average and Max/Min Hold).
[SENSe]:FVTime[:MEASview<y>]:COLor (on page 2132)	Sets or queries the color palette of three-dimensional graphs.
[SENSe]:FVTime[:MEASview<y>]:COLor:MAXimum (on page 2132)	Sets or queries the maximum value of the color axis in the Frequency versus Time measurement.
[SENSe]:FVTime[:MEASview<y>]:COLor:MINimum (on page 2133)	Sets or queries the minimum value of the color axis in the Frequency versus Time measurement.
[SENSe]:FVTime[:MEASview<y>]:COLor: REVerse[:STATe] (on page 2133)	Determines whether to reverse the color palette in the Frequency versus Time view.
[SENSe]:FVTime[:MEASview<y>]:FREQuency: CENTer (on page 2134)	Sets or queries the center frequency.
[SENSe]:FVTime[:MEASview<y>]:FREQuency: SPAN (on page 2135)	Sets or queries the frequency span.
[SENSe]:FVTime[:MEASview<y>]:FREQuency: STARt (on page 2136)	Sets or queries the measurement start frequency.

Command	Description
[SENSe]:FVTime[:MEASview<y>]:FREQuency: STEP (on page 2136)	Sets or queries the frequency step size.
[SENSe]:FVTime[:MEASview<y>]:FREQuency: STEP:AUTO (on page 2137)	Sets or queries whether to set the frequency step size automatically.
[SENSe]:FVTime[:MEASview<y>]:FREQuency: STOP (on page 2137)	Sets or queries the measurement stop frequency.
[SENSe]:FVTime[:MEASview<y>]:MAXTracepoints (on page 2138)	Sets or queries the maximum trace points.
[SENSe]:FVTime[:MEASview<y>]:OGRAm:COLor (on page 2138)	Sets or queries the ogram color scheme.
[SENSe]:FVTime[:MEASview<y>]:OGRAm:COLor: MAXimum (on page 2139)	Sets or queries the maximum value of the color axis in the ogram.
[SENSe]:FVTime[:MEASview<y>]:OGRAm:COLor: MINimum (on page 2140)	Sets or queries the minimum value of the color axis in the ogram.
[SENSe]:FVTime[:MEASview<y>]:OGRAm:COLor: REVerse (on page 2141)	Sets or queries reverse color scheme state.
[SENSe]:FVTime[:MEASview<y>]:PLOT (on page 2141)	Sets or queries the Frequency vs time display mode.
[:SENSe]:FVTime[:MEASview<y>]:SELEcted: SOURce (on page 2142)	Sets or queries the Source with respect to the Frequency versus Time Measurement View nuumber
[SENSe]:IQVTime subgroup	**RF I&Q versus Time measurement**
[SENSe]:IQVTime[:MEASview<y>]:CLEar:RESults (on page 2142)	Restarts multi-trace functions (Average and Max/Min Hold).
[SENSe]:IQVTime[:MEASview<y>]:FREQuency: CENTer (on page 2143)	Sets or queries the center frequency.
[SENSe]:IQVTime[:MEASview<y>]:FREQuency: SPAN (on page 2143)	Sets or queries the frequency span.
[SENSe]:IQVTime[:MEASview<y>]:FREQuency: STARt (on page 2144)	Sets or queries the measurement start frequency.
[SENSe]:IQVTime[:MEASview<y>]:FREQuency: STEP (on page 2145)	Sets or queries the frequency step size.
[SENSe]:IQVTime[:MEASview<y>]:FREQuency: STEP:AUTO (on page 2146)	Sets or queries whether to set the frequency step size automatically.
[SENSe]:IQVTime[:MEASview<y>]:FREQuency: STOP (on page 2146)	Sets or queries the measurement stop frequency.

Command	Description
[SENSe]:IQVTime[:MEASview<y>]:MAXTracepoints (on page 2147)	Sets or queries the maximum trace points.
[SENSe]:IQVTime[:MEASview<y>]:SELEcted: SOURce (on page 2147)	Sets or queries the Source with respect to the RF I&Q versus Time Measurement View number
[SENSe]:MCPower subgroup	**MCPR measurement**
[SENSe]:MCPower:AVERage (on page 2178)	Sets or queries how to average waveform.
[SENSe]:MCPower:AVERage:COUNt (on page 2178)	Sets or queries the number of waveforms for average.
[SENSe]:MCPower:{BANDwidth|BWIDth}[: RESolution] (on page 2179)	Sets or queries the resolution bandwidth (RBW).
[SENSe]:MCPower:{BANDwidth|BWIDth}[: RESolution]:ACTual? (on page 2179)	Queries the actual resolution bandwidth (RBW).
[SENSe]:MCPower:{BANDwidth|BWIDth}[: RESolution]:AUTO (on page 2180)	Sets or queries whether to set the RBW automatically.
[SENSe]:MCPower:{BANDwidth|BWIDth}:VIDeo (on page 2180)	Sets or queries the video bandwidth (VBW).
[SENSe]:MCPower:{BANDwidth|BWIDth}:VIDeo: STATe (on page 2180)	Sets or queries whether to enable the video bandwidth (VBW).
[SENSe]:MCPower:CHANnel:ADJacent:ADD (on page 2181)	Adds a pair of upper and lower adjacent channels.
[SENSe]:MCPower:CHANnel:ADJacent:DELete (on page 2182)	Deletes a selected adjacent channel.
[SENSe]:MCPower:CHANnel:ADJacent:LIMIts<x>: ENABle? (on page 2182)	Queries whether the adjacent channel is selected or not in the P25 MCPR display.
[SENSe]:MCPower:CHANnel:ADJacent:LIMIts<x> (on page 2182)	Sets or queries the selected adjacent upper and lower limit values in the P25 MCPR display.
[SENSe]:MCPower:CHANnel:FILTer (on page 2183)	Sets or queries the measurement filter.
[SENSe]:MCPower:CHANnel:MAIN:{BANDwidth| BWIDth} (on page 2183)	Sets or queries the frequency bandwidth of the main channels.
[SENSe]:MCPower:CHANnel:MAIN:COUNt (on page 2184)	Sets or queries the number of main channels.
[SENSe]:MCPower:CHANnel:MAIN:INACtive (on page 2184)	Makes a main channel inactive or queries the inactive main channels.
[SENSe]:MCPower:CHANnel:MAIN:SPACing (on page 2185)	Sets or queries the main channel spacing.
[SENSe]:MCPower:CHIPrate (on page 2185)	Sets or queries the chip rate.
[SENSe]:MCPower:CLEar:RESults (on page 2185)	Restarts the average trace.

Command	Description
[SENSe]:MCPower:FREQuency (on page 2186)	Sets or queries the center frequency.
[SENSe]:MCPower:FREQuency:STEP (on page 2186)	Sets or queries the frequency step size.
[SENSe]:MCPower:FREQuency:STEP:AUTO (on page 2187)	Sets or queries whether to set the frequency step size automatically.
[SENSe]:MCPower:NFLoor:STATe (on page 2187)	Sets or queries whether to enable the correction for noise floor.
[SENSe]:MCPower:POINts:COUNt (on page 2188)	Sets or queries the number of trace points.
[SENSe]:MCPower:RCHannels? (on page 2188)	Queries the power reference.
[SENSe]:MCPower:RCHannels:MAIN<x> (on page 2189)	Sets the power reference to the main channel with the index (<x>).
[SENSe]:MCPower:RCHannels:TOTal (on page 2189)	Sets the power reference to the total power of all the active channels.
[SENSe]:MCPower:RRCRolloff (on page 2190)	Sets or queries the filter parameter for the Root Raised Cosine filter.
[SENSe]:OBWidth subgroup	**Occupied Bandwidth measurement**
[SENSe]:OBWidth:AVERage (on page 2204)	Sets or queries whether to enable or disable averaging.
[SENSe]:OBWidth:AVERage:COUNt (on page 2204)	Sets or queries the number of measurements for averaging.
[SENSe]:OBWidth:{BANDwidth|BWIDth}: MEASurement (on page 2205)	Sets or queries the measurement bandwidth.
[SENSe]:OBWidth:{BANDwidth|BWIDth}[: RESolution] (on page 2205)	Sets or queries the resolution bandwidth (RBW).
[SENSe]:OBWidth:{BANDwidth|BWIDth}[: RESolution]:ACTual? (on page 2206)	Queries the actual resolution bandwidth (RBW).
[SENSe]:OBWidth:{BANDwidth|BWIDth}[: RESolution]:AUTO (on page 2206)	Sets or queries whether to set the RBW automatically.
[SENSe]:OBWidth:{BANDwidth|BWIDth}:VIDeo (on page 2207)	Sets or queries the video bandwidth (VBW).
[SENSe]:OBWidth:{BANDwidth|BWIDth}:VIDeo: STATe (on page 2207)	Sets or queries whether to enable the video bandwidth (VBW).
[SENSe]:OBWidth:CLEar:RESults (on page 2208)	Restarts the averaging.
[SENSe]:OBWidth:FREQuency:CENTer (on page 2209)	Sets or queries the center frequency.
[SENSe]:OBWidth:FREQuency:STEP (on page 2209)	Sets or queries the frequency step size.
[SENSe]:OBWidth:FREQuency:STEP:AUTO (on page 2210)	Determines whether to set the frequency step size automatically.
[SENSe]:OBWidth:PERCent (on page 2210)	Sets or queries the occupied bandwidth percent power.

Command	Description
[SENSe]:OBWidth:XDBLevel (on page 2211)	Sets or queries the x dB level.
[SENSe]:OFDM subgroup	**Orthogonal Frequency Division Multiplexing measurement**
[SENSe]:OFDM:CHANnel:ESTimation (on page 2213)	Sets or queries the channel estimation.
[SENSe]:OFDM:CHANnel[:BANDwidth|BWIDth] (on page 2211)	Sets or queries the channel bandwidth.
[SENSe]:OFDM:CHANnel[:BANDwidth|BWIDth]: SRATe? (on page 2212)	Returns the FFT sample rate.
[SENSe]:OFDM:CHANnel[:BANDwidth|BWIDth]: STATe (on page 2212)	Determines whether the channel bandwidth is on or off.
[SENSe]:OFDM:CONSte:DETermination (on page 2213)	Sets or queries the constellation determination.
[SENSe]:OFDM:CONSte:DETermination:MANual (on page 2213)	Sets or queries a manual constellation type.
[SENSe]:OFDM:FFT:LENGth? (on page 2219)	Returns the FFT length.
[SENSe]:OFDM:GUARd:INTerval (on page 2214)	Sets or queries the guard interval.
[SENSe]:OFDM:GUARd:INTerval:USER (on page 2214)	Sets or queries a user-defined guard interval.
[SENSe]:OFDM:PILot:TRACking:AMPLitude:STATe (on page 2215)	Determines whether to select the Amplitude pilot tracking.
[SENSe]:OFDM:PILot:TRACking:PHASe:STATe (on page 2215)	Determines whether to select the Phase pilot tracking.
[SENSe]:OFDM:PILot:TRACking:TIMing:STATe (on page 2216)	Determines whether to select the Timing pilot tracking.
[SENSe]:OFDM:RADix (on page 2216)	Sets the radix for the OFDM measurements.
[SENSe]:OFDM:SCARrier:SPACing (on page 2217)	Sets or queries the subcarrier spacing.
[SENSe]:OFDM:SCARrier:SPACing:STATe (on page 2217)	Determines whether the subcarrier spacing is on or off.
[SENSe]:OFDM:SCARriers (on page 2217)	Sets or queries the Subcarriers to display.
[SENSe]:OFDM:SCARriers:RANGe:STARt (on page 2218)	Sets or queries the subcarrier start range.
[SENSe]:OFDM:SCARriers:RANGe:STOP (on page 2218)	Sets or queries the subcarrier stop range.
[SENSe]:OFDM:SCARriers:SINGle:INDex (on page 2219)	Sets or queries the index of a single subcarrier.
[SENSe]:OFDM:STANdard (on page 2220)	Sets or queries the OFDM standard.

Command	Description
[SENSe]:OFDM:SWAP:IQ (on page 2220)	Determines whether or not to exchange I and Q data.
[SENSe]:OFDM:SYMBol:ANALysis:OFFSet (on page 2220)	Sets or queries the OFDM symbol analysis offset.
[SENSe]:OFDM:SYMBols (on page 2221)	Sets or queries the symbols to display.
[SENSe]:OFDM:SYMBols:MAX (on page 2221)	Sets or queries the maximum number of symbols to analyze.
[SENSe]:OFDM:SYMBols:MAX:STATe (on page 2222)	Enables or disables the maximum number of symbols to analyze.
[SENSe]:OFDM:SYMBols:RANGe:COUNt (on page 2222)	Sets or queries the number of symbols to display.
[SENSe]:OFDM:SYMBols:RANGe:STARt (on page 2223)	Sets or queries the symbols start range.
[SENSe]:OFDM:SYMBols:SINGle:INDex (on page 2223)	Sets or queries the symbol index
[SENSe]:OFDM:UNIT:FREQuency (on page 2223)	Specifies the frequency units.
[SENSe]:OFDM:UNIT:TIME (on page 2224)	Specifies the time units.
[SENSe]:PHVTime subgroup	**Phase versus Time measurement**
[SENSe]:PHVTime[:MEASview<y>]:CLEar:RESults (on page 2253)	Restarts multi-trace functions (Average and Max/Min Hold).
[SENSe]:PHVTime[:MEASview<y>]:COLor (on page 2253)	Sets or queries the color palette of three-dimensional graphs.
[SENSe]:PHVTime[:MEASview<y>]:COLor:MAXimum (on page 2254)	Sets or queries the maximum value of the color axis in the Phase versus Time measurement.
[SENSe]:PHVTime[:MEASview<y>]:COLor:MINimum (on page 2255)	Sets or queries the minimum value of the color axis in the Phase versus Time measurement.
[SENSe]:PHVTime[:MEASview<y>]:COLor: REVerse[:STATe] (on page 2255)	Determines whether to reverse the color palette in the Phase versus Time view.
[SENSe]:PHVTime[:MEASview<y>]:FREQuency: CENTer (on page 2256)	Sets or queries the center frequency.
[SENSe]:PHVTime[:MEASview<y>]:FREQuency: SPAN (on page 2256)	Sets or queries the frequency span.
[SENSe]:PHVTime[:MEASview<y>]:FREQuency: STARt (on page 2257)	Sets or queries the measurement start frequency.
[SENSe]:PHVTime[:MEASview<y>]:FREQuency: STEP (on page 2258)	Sets or queries the frequency step size.
[SENSe]:PHVTime[:MEASview<y>]:FREQuency: STEP:AUTO (on page 2258)	Sets or queries whether to set the frequency step size automatically.

Command	Description
[SENSe]:PHVTime[:MEASview<y>]:FREQuency: STOP (on page 2259)	Sets or queries the measurement stop frequency.
[SENSe]:PHVTime[:MEASview<y>]:MAXTracepoints (on page 2259)	Sets or queries the maximum trace points.
[SENSe]:PHVTime[:MEASview<y>]:OGRAm:COLor (on page 2260)	Sets or queries the Ogram color scheme.
[SENSe]:PHVTime[:MEASview<y>]:OGRAm:COLor:MAXimum (on page 2261)	Sets or queries the maximum value of the color axis in the ogram
[SENSe]:PHVTime[:MEASview<y>]:OGRAm:COLor:MINimum (on page 2262)	Sets or queries the minimum value of the color axis in the ogram.
[SENSe]:PHVTime[:MEASview<y>]:OGRAm:COLor:REVerse (on page 2262)	Sets or queries reverse color scheme state.
[SENSe]:PHVTime[:MEASview<y>]:PLOT (on page 2263)	Sets or queries the Phase vs Time display mode.
[SENSe]:PHVTime[:MEASview<y>]:SELEcted: SOURce (on page 2263)	Sets or queries the Source with respect to the Phaseversus Time Measurement View nuumber
[SENSe]:PNOise subgroup (Option 11 only)	**Phase noise measurement**
[SENSe]:PNOise:AVERage:COUNt (on page 2265)	Sets or queries the number of traces to combine for averaging.
[SENSe]:PNOise:AVERage:ENABle (on page 2266)	Sets or queries whether to enable or disable averaging trace.
[SENSe]:PNOise:CARRier:FREQuency:TRACk (on page 2266)	Sets or queries whether to enable tracking the carrier frequency.
[SENSe]:PNOise:CARRier:THReshold (on page 2266)	Sets or queries the threshold level to detect the carrier.
[SENSe]:PNOise:CLEar:RESults (on page 2267)	Restarts the average process.
[SENSe]:PNOise:FREQuency:INTegration:OFFSet: STARt (on page 2267)	Sets or queries the start offset frequency for integration.
[SENSe]:PNOise:FREQuency:INTegration:OFFSet: STOP (on page 2268)	Sets or queries the stop offset frequency for integration.
[SENSe]:PNOise:FREQuency:PLOT:OFFSet:STARt (on page 2268)	Sets or queries the start offset frequency for plot.
[SENSe]:PNOise:FREQuency:PLOT:OFFSet:STOP (on page 2268)	Sets or queries the stop offset frequency for plot.
[SENSe]:PNOise:OPTimization (on page 2269)	Sets or queries the method of optimization.
[SENSe]:PULSe subgroup	**Pulsed RF measurements**

Command	Description
[SENSe]:PULSe[:MEASview<y>]:ANALyze:FDOMain:MEASurement:TIME:ALENgth (on page 2270)	Sets or queries the absolute amount of time included in the Frequency-Domain Linearity measurements.
[SENSe]:PULSe[:MEASview<y>]:ANALyze:FDOMain:MEASurement:TIME:METHod (on page 2271)	Sets or queries the measurement time method for the Frequency-Domain Linearity measurements.
[SENSe]:PULSe[:MEASview<y>]:ANALyze:FDOMain:MEASurement:TIME:RLENgth (on page 2271)	Sets or queries the relative amount of time included in the Frequency-Domain Linearity measurements.
[SENSe]:PULSe[:MEASview<y>]:ANALyze:FDOMain:MEASurement:TIME:RLEVel (on page 2272)	Sets or queries the Frequency-Domain Linearity measurement reference level.
[SENSe]:PULSe[:MEASview<y>]:ANALyze:FDOMain:MEASurement:TIME:STARt (on page 2273)	Sets or queries the start time for the Frequency-Domain Linearity measurements
[SENSe]:PULSe[:MEASview<y>]:ANALyze:IRESponse:CORRection:AMPLitude[:STATe] (on page 2273)	Enables or disables the impulse response measurement amplitude corrections for window and for delay of the highest side lobe.
[SENSe]:PULSe[:MEASview<y>]:ANALyze:IRESponse:KOTime (on page 2274)	Sets or queries the impulse response measurement Keep-out time.
[SENSe]:PULSe[:MEASview<y>]:ANALyze:IRESponse:MEASurement:TIME:ALENgth (on page 2274)	Sets or queries the absolute amount of time included in impulse response measurements.
[SENSe]:PULSe[:MEASview<y>]:ANALyze:IRESponse:MEASurement:TIME:METHod (on page 2275)	Sets or queries the impulse response measurement time method.
[SENSe]:PULSe[:MEASview<y>]:ANALyze:IRESponse:MEASurement:TIME:RLENgth (on page 2276)	Sets or queries the relative amount of time included in Impulse Response measurements.
[SENSe]:PULSe[:MEASview<y>]:ANALyze:IRESponse:MEASurement:TIME:RLEVel (on page 2276)	Sets or queries the measurement reference level for Impulse Response measurements.
[SENSe]:PULSe[:MEASview<y>]:ANALyze:IRESponse:MEASurement:TIME:STARt (on page 2277)	Sets or queries the start time for the Impulse Response measurements.
[SENSe]:PULSe[:MEASview<y>]:ANALyze:LEVel (on page 2277)	Sets or queries how to determine the 50% level.
[SENSe]:PULSe[:MEASview<y>]:ANALyze:LEVel: FIFTy (on page 2278)	Sets or queries how to determine the 50% level.
[SENSe]:PULSe[:MEASview<y>]:ANALyze:LEVel: HUNDred (on page 2278)	Sets or queries how to determine the 50% level.
[SENSe]:PULSe[:MEASview<y>]:ANALyze:OVERshoot:MEASurement:TIME:ALENgth (on page 2280)	Sets or queries the absolute amount of time included in the Overshoot measurements.
[SENSe]:PULSe[:MEASview<y>]:ANALyze:OVERshoot:MEASurement:TIME:METHod (on page 2281)	Sets or queries the measurement time method for the Overshoot measurements.
[SENSe]:PULSe[:MEASview<y>]:ANALyze:OVERshoot:MEASurement:TIME:RLENgth (on page 2281)	Sets or queries the relative amount of time included in the Overshoot measurements.

Command	Description
[SENSe]:PULSe[:MEASview<y>]:ANALyze: OVERshoot:MEASurement:TIME:RLEVel (on page 2282)	Sets or queries the measurement reference level for the Overshoot measurements.
[SENSe]:PULSe[:MEASview<y>]:ANALyze: OVERshoot:MEASurement:TIME:STARt (on page 2283)	Sets or queries the start time for the Overshoot measurements
[SENSe]:PULSe[:MEASview<y>]:ANALyze: PMLocation (on page 2283)	Sets or queries the phase measurement location.
[SENSe]:PULSe[:MEASview<y>]:ANALyze: POINt:LOCation (on page 2284)	Sets or queries the point location method.
[SENSe]:PULSe[:MEASview<y>]:ANALyze:RFALl (on page 2284)	Sets or queries the threshold levels to measure the rise/fall time.
[SENSe]:PULSe[:MEASview<y>]:ANALyze:RIPPle (on page 2285)	Sets or queries the ripple portion of the pulse top.
[SENSe]:PULSe[:MEASview<y>]:CARRier: FREQuency? (on page 2285)	Queries the carrier frequency.
[SENSe]:PULSe[:MEASview<y>]:CARRier:OFFSet (on page 2286)	Sets or queries the carrier frequency offset.
[SENSe]:PULSe[:MEASview<y>]:CARRier:SEARch (on page 2287)	Sets or queries how to detect the carrier.
[SENSe]:PULSe[:MEASview<y>]:DETect: MEASurement (on page 2292)	Sets or queries whether to set the maximum number of pulses.
[SENSe]:PULSe[:MEASview<y>]:DETect:NUMBer (on page 2293)	Sets or queries the maximum number of pulses to detect.
[SENSe]:PULSe[:MEASview<y>]:DETect:NUMBer: THReshold (on page 2293)	Sets or queries the power threshold to detect pulses.
[SENSe]:PULSe[:MEASview<y>]:DETect:TIMe[: THReshold] (on page 2294)	Sets or queries the minimum off-time between pulses.
[SENSe]:PULSe[:MEASview<y>]:FILTer: {BANDwidth| BWIDth} (on page 2295)	Sets or queries the filter bandwidth for the Gaussian filter.
[SENSe]:PULSe[:MEASview<y>]:FILTer: MEASurement (on page 2295)	Sets or queries the measurement filter.
[SENSe]:PULSe[:MEASview<y>]:FREFerence:AUTO (on page 2296)	Determines whether to estimate the frequency reference automatically.
[SENSe]:PULSe[:MEASview<y>]:FREFerence: CSLope (on page 2297)	Sets or queries the Chirp Slope.
[SENSe]:PULSe[:MEASview<y>]:FREFerence: CSLope:AUTO (on page 2297)	Sets or queries enabling automatic Chirp Slope estimating.

Command	Description
[SENSe]:PULSe[:MEASview<y>]:FREFerence: OFFSet (on page 2298)	Sets or queries the frequency reference offset.
[SENSe]:PULSe[:MEASview<y>]:MODulation:TYPE (on page 2298)	Sets or queries the modulation type.
[SENSe]:PULSe[:MEASview<y>]:SIGNal:TYPE (on page 2303)	Sets or queries the signal type in the pulsed RF measurements.
[SENSe]:PULSe[:MEASview<y>]:STATistics:HISTogram:ORDinate (on page 2304)	Sets or queries the Histogram Y unit.
[SENSe]:PULSe[:MEASview<y>]:STATistics:HISTogram:POINts (on page 2304)	Sets or queries the number of bins in the Histogram plot.
[SENSe]:PULSe[:MEASview<y>]:STATistics:HISTogram:POINts:AUTO (on page 2305)	Sets or queries enabling automatic Histogram plot bins count.
[SENSe]:REANalyze subgroup	**Spectrogram measurement**
[SENSe]:REANalyze (on page 2307)	Sets all measurements to reanalyze the current acquisition record.
[SENSe]:REANalyze:ACQuisition:SETTings? (on page 2307)	Queries the settings of all acquisitions in acquisition history.
[SENSe]:REANalyze:ALL	Replay all items in the current acquisition history selection.
[SENSe]:REANalyze:ALL:LOOP (on page 2307)	Replay all selected records/frames in a continuous loop.
[SENSe]:REANalyze:CURRent (on page 2308), [SENSe]:REANalyze:CURRent (on page 2308)	Replay the currently selected acquisition (or frame).
[SENSe]:REANalyze:CURRent:ACQuisition? (on page 2308)	Queries the Acquisition record identifier (ID) of the currently selected acquisition (or frame).
[SENSe]:REANalyze:CURRent:FRAMe? (on page 2308)	Queries the Frame identifier (ID) of the currently selected acquisition (or frame).
[SENSe]:REANalyze:CURRent:TIMestamp? (on page 2309)	Queries the time stamp of the currently selected acquisition (or frame).
[SENSe]:REANalyze:DELete:DATA (on page 2309)	Deletes all acquisition data.
[SENSe]:REANalyze:FIRSt (on page 2309)	Replays the first item in the current acquisition history selection.
[SENSe]:REANalyze:LAST (on page 2309)	Replays the last item in the current acquisition history selection.
[SENSe]:REANalyze:NEXT (on page 2310)	Replays the next item in the current acquisition history selection.
[SENSe]:REANalyze:PAUSe (on page 2310)	Sets or queries the reanalysis pause state.
[SENSe]:REANalyze:PREVious (on page 2310)	Replays the previous item in the current Replay selection.

Command	Description
[SENSe]:REANalyze:SELect:ACQuisition:DATA: TYPE (on page 2311)	Sets or queries the acquisition data type.
[SENSe]:REANalyze:SELect:ACQuisition:FIRSt? (on page 2311)	Queries the first selectable acquisition history Start or Stop value.
[SENSe]:REANalyze:SELect:ACQuisition:FIRSt: TIMestamp? (on page 2311)	Queries the time stamp associated with the first selectable acquisition history Start or Stop value.
[SENSe]:REANalyze:SELect:ACQuisition:LAST? (on page 2312)	Queries the last selectable acquisition history Start or Stop valu
[SENSe]:REANalyze:SELect:ACQuisition:LAST: TIMestamp? (on page 2312)	Queries the time stamp associated with the last selectable acquisition history Start or Stop value
[SENSe]:REANalyze:SELect:ACQuisition:STARt (on page 2312)	Sets or queries the Acquisition Start value in the acquisition history selection.
[SENSe]:REANalyze:SELect:ACQuisition:STARt: TIMestamp? (on page 2313)	Queries the time stamp associated with the current Acquisition Start selection
[SENSe]:REANalyze:SELect:ACQuisition:STOP (on page 2313)	Sets or queries the Acquisition Stop value in the acquisition history selection.
[SENSe]:REANalyze:SELect:ACQuisition:STOP: TIMestamp? (on page 2313)	Queries the time stamp associated with the current Acquisition Stop selection.
[SENSe]:REANalyze:SELect:ALL (on page 2314)	Selects all acquisitions and frames in the current acquisition history.
[SENSe]:REANalyze:SELect:FRAMe:FIRSt? (on page 2314)	Queries the first available Frame Start or Stop value available for acquisition history selection.
[SENSe]:REANalyze:SELect:FRAMe:FIRSt: TIMestamp? (on page 2314)	Queries the time stamp associated with the first available frame of the first available Acquisition Start or Stop value
[SENSe]:REANalyze:SELect:FRAMe:LAST? (on page 2315)	Queries the last available Frame Start or Stop value available for acquisition history selection.
[SENSe]:REANalyze:SELect:FRAMe:LAST: TIMestamp? (on page 2315)	Queries the time stamp associated with the last available frame of the last available Acquisition Start or Stop value
[SENSe]:REANalyze:SELect:FRAMe:STARt (on page 2315)	Sets or queries the Frame Start value in the acquisition history selection.
[SENSe]:REANalyze:SELect:FRAMe:STARt: TIMestamp? (on page 2316)	Queries the time stamp associated with the current Frame Start selection.
[SENSe]:REANalyze:SELect:FRAMe:STOP (on page 2316)	Sets or queries the frame stop value in acquisition history selection.
[SENSe]:REANalyze:SELect:FRAMe:STOP: TIMestamp? (on page 2316)	Queries the time stamp associated with the current frame stop selection.

Command	Description
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:ALL (on page 2317)	Selects all spectra acquisitions and frames in the currently selected set.
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:CURRent:ACQuisition? (on page 2317)	Queries the acquisition record identifier (ID) of the currently selected acquisition.
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:CURRent:TIMestamp? (on page 2317)	Queries the time stamp of the currently selected acquisition.
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition: FIRSt? (on page 2318)	Queries the first selectable acquisition start or stop value in the currently selected set.
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:FIRSt:TIMestamp? (on page 2318)	Queries the time stamp of the first selectable acquisition history start or stop value in the currently selected acquisition set.
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition: LAST? (on page 2319)	Queries the last selectable acquisition start or stop value in the currently selected set.
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:LAST:TIMestamp? (on page 2319)	Queries the time stamp of the last selectable acquisition start or stop value in the currently selected set.
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition: STARt (on page 2319)	Sets or queries the acquisition start value in the currently selected set.
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:STARt:TIMestamp? (on page 2320)	Queries the time stamp of the current acquisition start selection.
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition: STOP (on page 2320)	Sets or queries the acquisition stop value in the currently selected set.
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:STOP:TIMestamp? (on page 2321)	Queries the time stamp of the current acquisition stop selection.
[SENSe]:REANalyze:SPEed (on page 2321)	Sets or queries the time delay value inserted between reanalysis of frames or acquisitions.
[SENSe]:REANalyze:STOP (on page 2321)	Stops the current reanalysis.
[SENSe]:REANalyze:TIMestamp:DECimal (on page 2322)	Sets or queries the number of values to the right of the decimal point to include in all time stamps.
[SENSe]:SEM subgroup	**Spectrum Emissions Mask measurements**
[SENSe]:SEM:{BANDwidth|BWIDth}[:RESolution] (on page 2324)	Sets or queries the resolution bandwidth (RBW).
[SENSe]:SEM:{BANDwidth|BWIDth}:VIDeo (on page 2324)	Sets or queries the video bandwidth (VBW).
[SENSe]:SEM:{BANDwidth|BWIDth}:VIDeo:STATe (on page 2325)	Sets or queries the state of the video bandwidth (VBW) of the reference channel.
[SENSe]:SEM:CHANnel:FILTer (on page 2325)	Sets or queries the reference Channel Filter.

Command	Description
[SENSe]:SEM:CHANnel:IBANdwidth (on page 2325)	Sets or queries the reference channel integration bandwidth.
[SENSe]:SEM:CHANnel:WIDTh (on page 2326)	Sets or queries the reference Channel Filter bandwidth.
[SENSe]:SEM:CHIPrate (on page 2326)	Sets or queries the Chip rate.
[SENSe]:SEM:CLEar:RESults (on page 2327)	Clears the results in the Spectral Emissions Mask measurement.
[SENSe]:SEM:FREQuency (on page 2327)	Sets or queries the measurement frequency.
[SENSe]:SEM:FREQuency:SPAN? (on page 2328)	Returns the frequency span.
[SENSe]:SEM:MFILter (on page 2328)	Sets or queries the shape of the filter.
[SENSe]:SEM:MODE:REALtime (on page 2328)	Sets or queries the Real-time SEM measurement mode.
[SENSe]:SEM:MTYPe (on page 2329)	Sets or queries the measurement type.
[SENSe]:SEM:OFFSet:DETection (on page 2329)	Sets or queries the detection settings for all offsets.
[SENSe]:SEM:OFFSet<x>:BANDwidth:FILTer? (on page 2330)	Returns the bandwidth filter setting.
[SENSe]:SEM:OFFSet<x>:BANDwidth[:RESolution] (on page 2330)	Sets or queries the resolution bandwidth.
[SENSe]:SEM:OFFSet<x>:BANDwidth:RESolution: COUNt (on page 2331)	Sets or queries the count of Resolution Bandwidths (RBWs).
[SENSe]:SEM:OFFSet<x>:BANDwidth:VIDeo (on page 2331)	Sets or queries the video bandwidth setting.
[SENSe]:SEM:OFFSet<x>:BANDwidth:VIDeo:STATe (on page 2332)	Sets or queries the state of the video bandwidth.
[SENSe]:SEM:OFFSet<x>:FREQuency:STARt (on page 2332)	Sets or queries the start frequency.
[SENSe]:SEM:OFFSet<x>:FREQuency:STOP (on page 2333)	Sets or queries the stop frequency.
[SENSe]:SEM:OFFSet<x>:LIMIT:ABSolute:SAME (on page 2333)	Sets or queries the state of the limits for the absolute start and stop settings.
[SENSe]:SEM:OFFSet<x>:LIMIT:ABSolute:STARt (on page 2333)	Sets or queries the absolute power start limit.
[SENSe]:SEM:OFFSet<x>:LIMIT:ABSolute:STOP (on page 2334)	Sets or queries the absolute power stop limit.
[SENSe]:SEM:OFFSet<x>:LIMIT:MASK (on page 2334)	Sets or queries the limit mask setting.

Command	Description
[SENSe]:SEM:OFFSet<x>:LIMIT:RELative:SAME (on page 2335)	Sets or queries the state of the limits for the relative start and stop settings
[SENSe]:SEM:OFFSet<x>:LIMIT:RELative:STARt (on page 2335)	Sets or queries the relative power start limit.
[SENSe]:SEM:OFFSet<x>:LIMIT:RELative:STOP (on page 2336)	Sets or queries the relative power stop limit.
[SENSe]:SEM:OFFSet<x>:LIMIT:SIDE (on page 2336)	Sets or queries the side of the offset limit.
[SENSe]:SEM:OFFSet<x>:STATe (on page 2336)	Determines whether to display the specified row of the Offsets & Limits table.
[SENSe]:SEM:POINts:COUNt (on page 2337)	Sets or queries the maximum number of trace points.
[SENSe]:SEM:PRESet (on page 2337)	Defines the standard protocol settings for the measurement.
[SENSe]:SEM:REFerence:AUTO (on page 2338)	Determines whether to automatically update the Power Reference.
[SENSe]:SEM:REFerence:DETection (on page 2338)	Sets or queries the reference channel detection setting.
[SENSe]:SEM:REFerence[:MAGNitude] (on page 2339)	Sets or queries the power magnitude.
[SENSe]:SEM:RRCRolloff (on page 2339)	Sets or queries the filter parameter (roll-off ratio).
[SENSe]:SEM:STEP:AUTO (on page 2340)	Determines whether to automatically update the center frequency.
[SENSe]:SEM:STEP[:MAGNitude] (on page 2340)	Sets or queries the step magnitude.
[SENSe]:SEM:WIDTh:DEFinition (on page 2341)	Sets or queries the offset definition.
[SENSe]:SGRam subgroup	**Spectrogram measurements**
[SENSe]:SGRam[:MEASview<y>]:{BANDwidth| BWIDth}:RESolution (on page 2341)	Sets or queries the resolution bandwidth (RBW).
[SENSe]:SGRam[:MEASview<y>]:{BANDwidth| BWIDth} [:RESolution]: ACTual? (on page 2342)	Queries the actual resolution bandwidth (RBW).
[SENSe]:SGRam[:MEASview<y>]:{BANDwidth| BWIDth} [:RESolution]:AUTO (on page 2342)	Sets or queries whether to set the RBW automatically.
[SENSe]:SGRam[:MEASview<y>]:{BANDwidth| BWIDth} [:RESolution]:MODE (on page 2343)	Sets or queries whether to enable or disable the RBW processing.
[SENSe]:SGRam[:MEASview<y>]:{BANDwidth| BWIDth}:VIDeo (on page 2343)	Sets or queries the video bandwidth (VBW).
[SENSe]:SGRam[:MEASview<y>]:{BANDwidth| BWIDth}:VIDeo:STATe (on page 2344)	Sets or queries whether to enable the video bandwidth (VBW).

Command	Description
[SENSe]:SGRam[:MEASview<y>]:COLor (on page 2345)	Sets or queries the color palette of three-dimensional graphs.
[SENSe]:SGRam[:MEASview<y>]:COLor:MAXimum (on page 2346)	Sets or queries the maximum value of the color axis.
[SENSe]:SGRam[:MEASview<y>]:COLor:MINimum (on page 2347)	Sets or queries the minimum value of the color axis.
[SENSe]:SGRam[MEASview<y>]:COLor:REVerse (on page 2347)	Sets or queries reverse color scheme state.
[SENSe]:SGRam[:MEASview<y>]:FFT:WINDow (on page 2348)	Sets or queries the FFT window.
[SENSe]:SGRam[:MEASview<y>]:FILTer[:SHAPe] (on page 2349)	Sets or queries the filter shape.
[SENSe]:SGRam[:MEASview<y>]:FREQuency: CENTer (on page 2350)	Sets or queries the center frequency.
[SENSe]:SGRam[:MEASview<y>]:FREQuency: SPAN (on page 2351)	Sets or queries the frequency span.
[SENSe]:SGRam[:MEASview<y>]:FREQuency: SPAN:BANDwidth[:RESolution]:RATio (on page 2351)	Sets or queries the ratio of span to RBW.
[SENSe]:SGRam[:MEASview<y>]:FREQuency: SPAN:MAXimum (on page 2352)	Sets the frequency range to the maximum real-time span.
[SENSe]:SGRam[:MEASview<y>]:FREQuency: STARt (on page 2352)	Sets or queries the measurement start frequency.
[SENSe]:SGRam[:MEASview<y>]:FREQuency: STEP (on page 2353)	Sets or queries the frequency step size.
[SENSe]:SGRam[:MEASview<y>]:FREQuency: STEP:AUTO (on page 2354)	Sets or queries whether to set the frequency step size automatically.
[SENSe]:SGRam[:MEASview<y>]:FREQuency: STOP (on page 2354)	Sets or queries the measurement stop frequency.
[SENSe]:SGRam[:MEASview<y>]TIME[:SCALe]: MODe (on page 2355)	Sets or queries the vertical mode.
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]: OVERlap:PERCent (on page 2356)	Sets or queries the Overlap in percent.
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]: PER:DIVision (on page 2356)	Sets or queries the vertical Time/division value.
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]: PER:UPDate:MINutes (on page 2359)	Sets or queries the Spectrum Monitor mode Time/Update value for minutes.

Command	Description
[SENSe]:SGRam[:MEASview<y>]:WATerfall:ENABle (on page 2361)	Sets or queries the Waterfall (3-D spectrogram) display.
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:PER:UPDate:SEConds (on page 2359)	Sets or queries the Spectrum Monitor mode Time/Update value
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:SPECtrums:PERLine? (on page 2360)	Queries the Spectrums/Line value.
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:STARt:DIVisions (on page 2360)	Sets or queries the Vertical offset or Position in divisions.
[SENSe]:SGRam[:MEASview<y>]:WATerfall: DIRection (on page 2361)	Sets or queries the direction of the Waterfall display.
[SENSe]:SGRam[:MEASview<y>]:WATerfall:Y: AUTO (on page 2362)	Rescales the height axis automatically to fit the waveform to the screen in the 3-D Waterfall display.
[SENSe]:SGRam[:MEASview<y>]:WATerfall:Y: OFFSet (on page 2362)	Sets or queries the height offset (the value at the top edge of the vertical axis) of the 3-D Waterfall display.
[SENSe]:SGRam[:MEASview<y>]:WATerfall:Y: RESet (on page 2363)	Resets the height scale of the 3-D Waterfall display to the default values.
[SENSe]:SGRam[:MEASview<y>]:WATerfall: Y[:SCALe] (on page 2363)	Sets or queries the height (vertical) range of the Waterfall (3-D spectrogram) display.
[SENSE]:SIGNalvu subgroup	**Signalvu measurement**
[:SENSe]:SIGNalvu:ACQuisition:PER:CHANnel<x>:COUPling (on page 2369)	Sets or queries the per channel coupling of the measurement.
[:SENSe]:SIGNalvu:ACQuisition:PER:CHANnel<x>: DESKew (on page 2370)	Sets or queries the per channel deskew of the measurement.
[:SENSe]:SIGNalvu:ACQuisition:PER:CHANnel<x>: EXTattn (on page 2370)	Sets or queries the per channel external attenuation of the measurement.
[:SENSe]:SIGNalvu:ACQuisition:PER:CHANnel<x>:INSTrument:BWIDth (on page 2371)	Queries the per channel instrument bandwidth of the measurement.
[:SENSe]:SIGNalvu:ACQuisition:PER:CHANnel<x>: REFlevel (on page 2371)	Sets or queries per channel reference level of the measurement.
[:SENSe]:SIGNalvu:ACQuisition:PER:CHANnel<x>:TERMination (on page 2372)	Sets or queries the per channel Termination of the measurement.
[SENSe]:SIGNalvu:ACQuisition:CONTrol:SOURce<x>:STATe (on page 2365)	Sets or queries the Source checkbox state of measurement source for RF, I and Q, Diff I and Q data type.
[SENSe]:SIGNalvu:ACQuisition:GLOBal:REFlevel (on page 2368)	Sets or queries the Global Reference level of the measurement source.

Command	Description
[SENSe]:SIGNalvu:ACQuisition:INPuttype:REFlevel (on page 2368)	Sets or queries the input type of ref level of the measurement source.
[SENSe]:SIGNalvu:ACQuisition:NULL:IQ:OFFSet: CLEAR (on page 2369)	To clear Null IQ offset values that were calculated using Calibrate IQ, use the above PI command.
[SENSe]:SIGNalvu:ACQuisition:SELEcted:CHANnel: TYPE (on page 2373)	Sets or queries the selected channel type of measurement.
[SENSe]:SIGNalvu:ACQuisition:CONTrol:HORizontal (on page 2364)	Sets or queries the Acquisition or Horizontal settings. When the checkbox is checked, the SignalVu controls horizontal settings of the oscilloscope.
[SENSe]:SIGNalvu:ACQuisition:CONTrol:SAMPle: RATE (on page 2365)	Sets or queries the Sample rate when the checkbox is checked, the SignalVu controls Sample rate of the oscilloscope.
[SENSe]:SIGNalvu:ACQuisition:CONTrol:TRIGger: POSition (on page 2366)	Sets or queries the Trigger position, when checkbox is checked, the SignalVu controls Trigger position of the oscilloscope.
[SENSe]:SIGNalvu:ACQuisition:CONTrol:VERTical (on page 2366)	Sets or queries the Vertical settings, when the checkbox is checked, the SignalVu controls the vertical settings of the oscilloscope.
[SENSe]:SIGNalvu:ACQuisition:DIGitizer:SAMPle: RATe (on page 2367)	Sets or queries the oscilloscope sample rate from the Acquisition control panel osilloscope settings of SignalVu.
[SENSe]:SIGNalvu:ACQuisition:DIGitizer:SAMPle: RATE:AUTo (on page 2367)	Sets the oscilloscope sample rate automatically or manually from the Acquisition control panel oscilloscope settings of SignalVu.
[SENSe]:SIGNalvu:ACQuisition:IQ:CALibrate (on page 2369)	Starts the IQ calibration.
[SENSe]:SIGNalvu:ACQuisition:RESet (on page 2372)	Clears the acquisition data and restarts the SignalVu acquisitions from the oscilloscope.
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]: CALIbrate:IQ (on page 2373)	For performing null IQ offset operation for each source use the above-mentioned PI command. It will be effective from the next acquisition.
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]: CHANnel:I (on page 2374)	Sets or queries the channel of measurement source for I data type.
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]: CHANnel:IMINus (on page 2375)	Sets or queries the channel of measurement source for IMinus data type.
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]: CHANnel:IPLus (on page 2375)	Sets or queries the channel of measurement source for IPlus data type.
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]: CHANnel:Q (on page 2374)	Sets or queries the channel of measurement source for Q data type.
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]: CHANnel:QMINus (on page 2376)	Sets or queries the channel of measurement source for QMinus data type.
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]: CHANnel:QPLus (on page 2376)	Sets or queries the channel of measurement source for QPlus data type.

Command	Description
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]: CHANnel[:RF] (on page 2373)	Sets or queries the channel of measurement source for RF data type.
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]: DOWNconverter:INJEctionside (on page 2377)	Sets or queries the downconverter injection side.
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]: DOWNconverter:INSTrument:BANDwidth:MAXFreq (on page 2377)	Sets or queries the maximum range of instrument bandwidth.
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]: DOWNconverter:INSTrument:BANDwidth:MINFreq (on page 2378)	Sets or queries the minimum range of instrument bandwidth.
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]: DOWNconverter:LOFReq (on page 2378)	Sets or queries the downconverter low frequency.
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]: DOWNconverter:STATe (on page 2379)	Sets or queries the downconverter state.
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]: PERSrc:REFlevel (on page 2379)	Sets or queries the Reference level Per source.
[SENSe]:SPECtrum subgroup	**Spectrum measurement**
[SENSe]:SPECtrum[:MEASview<y>]: {BANDwidth|BWIDth} [:RESolution] (on page 2380)	Sets or queries the resolution bandwidth (RBW).
[SENSe]:SPECtrum[:MEASview<y>]: {BANDwidth|BWIDth}[: RESolution]:ACTual? (on page 2380)	Queries the actual resolution bandwidth (RBW).
[SENSe]:SPECtrum[:MEASview<y>]: {BANDwidth|BWIDth} [:RESolution]:AUTO (on page 2381)	Sets or queries whether to set the RBW automatically.
[SENSe]:SPECtrum[:MEASview<y>]: {BANDwidth|BWIDth} [:RESolution]:MODE (on page 2381)	Sets or queries whether to enable or disable the RBW process.
[SENSe]:SPECtrum[:MEASview<y>]: {BANDwidth| BWIDth}:VIDeo (on page 2382)	Sets or queries the video bandwidth (VBW).
[SENSe]:SPECtrum[:MEASview<y>]: {BANDwidth| BWIDth}:VIDeo:STATe (on page 2383)	Sets or queries whether to enable the video bandwidth (VBW).
[SENSe]:SPECtrum[:MEASview<y>]:CLEar:RESults (on page 2383)	Restarts multi-trace functions (Average and Max/Min Hold).
[SENSe]:SPECtrum[:MEASview<y>]:COLor (on page 2383)	Sets or queries the color palette of three-dimensional graphs.
[SENSe]:SPECtrum[:MEASview<y>]:COLor:MAXimum (on page 2384)	Sets or queries the maximum value of the color axis in the Spectrum measurement.
[SENSe]:SPECtrum[:MEASview<y>]:COLor:MINimum (on page 2385)	Sets or queries the minimum value of the color axis in the Spectrum measurement.

Command	Description
[SENSe]:SPECtrum[:MEASview<y>]:COLor:REVerse[:STATe] (on page 2385)	Determines whether to reverse the color palette in the Spectrum view.
[SENSe]:SPECtrum[:MEASview<y>]:FFT:WINDow (on page 2386)	Sets or queries the FFT window.
[SENSe]:SPECtrum[:MEASview<y>]:FILTer[:SHAPe] (on page 2387)	Sets or queries the filter shape.
[SENSe]:SPECtrum[:MEASview<y>]:FREQuency: CENTer (on page 2387)	Sets or queries the center frequency.
[SENSe]:SPECtrum[:MEASview<y>]:FREQuency: SPAN (on page 2388)	Sets or queries the frequency span.
[SENSe]:SPECtrum[:MEASview<y>]:FREQuency: SPAN:BANDwidth[: RESolution]:RATio (on page 2388)	Sets or queries the ratio of span to RBW.
[SENSe]:SPECtrum[:MEASview<y>]:FREQuency: STARt (on page 2389)	Sets or queries the measurement start frequency.
[SENSe]:SPECtrum[:MEASview<y>]:FREQuency: STEP (on page 2390)	Sets or queries the frequency step size.
[SENSe]:SPECtrum[:MEASview<y>]:FREQuency: STEP:AUTO (on page 2390)	Sets or queries whether to set the frequency step size automatically.
[SENSe]:SPECtrum[:MEASview<y>]:FREQuency: STOP (on page 2391)	Sets or queries the measurement stop frequency.
[SENSe]:SPECtrum[:MEASview<y>]:LENGth (on page 2391)	Sets or queries the spectrum length.
[SENSe]:SPECtrum[:MEASview<y>]:LENGth: ACTual? (on page 2392)	Queries the actual spectrum length.
[SENSe]:SPECtrum[:MEASview<y>]:LENGth:AUTO (on page 2392)	Sets or queries the spectrum length mode (automatic or manual).
[SENSe]:SPECtrum[:MEASview<y>]:MAX:SPAN (on page 2393)	Sets the frequency span to the maximum span.
[SENSe]:SPECtrum[:MEASview<y>]:POINts:COUNt (on page 2393)	Sets or queries the number of sample points on the signal spectrum.
[SENSe]:SPECtrum[:MEASview<y>]:STARt (on page 2394)	Sets or queries the spectrum offset time.
[SENSe]:SPECtrum[:MEASview<y>]:STARt:AUTO (on page 2394)	Sets or queries the spectrum offset mode (automatic or manual).
[SENSe]:SPECtrum[:MEASview<y>]:TIMe:MODe (on page 2395)	Sets or queries the spectrum time parameter mode (automatic or manual).
[SENSe]:SPURious subgroup	**Spurious measurement**

Command	Description
[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth} (on page 2395)	Sets or queries the channel width for the carrier as power reference.
[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth}:INTegration (on page 2396)	Sets or queries the integration bandwidth to calculate the carrier power.
[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth}[:RESolution] (on page 2396)	Sets or queries the resolution bandwidth to measure the carrier power.
[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth}[:RESolution]:AUTO (on page 2397)	Sets or queries whether to set the resolution bandwidth automatically.
[SENSe]:SPURious:CARRier:DETection (on page 2397)	Sets or queries the carrier detection method.
[SENSe]:SPURious:CARRier:FREQuency (on page 2398)	Sets or queries the carrier frequency.
[SENSe]:SPURious:CARRier:THReshold (on page 2398)	Sets or queries the threshold level to detect the carrier.
[SENSe]:SPURious:CLEar:RESults (on page 2399)	Restarts multi-trace functions (Average and Max Hold).
[SENSe]:SPURious[:FREQuency]:OVERlap? (on page 2399)	Queries whether any of the frequency ranges (A to T) overlap.
[SENSe]:SPURious:LIST (on page 2400)	Sets or queries how to list the spurious signals.
[SENSe]:SPURious:MODE (on page 2400)	Sets or queries the frequency range mode (Multi or Single).
[SENSe]:SPURious:OPTimization (on page 2401)	Sets or queries the method of optimization.
[SENSe]:SPURious:POINts:COUNt (on page 2401)	Sets or queries the trace point count for the frequency range.
[SENSe]:SPURious:RANGe<x>:BANDwidth:VIDeo (on page 2402)	Sets or queries the VBW for the specified frequency range.
[SENSe]:SPURious:RANGe<x>:BANDwidth:VIDeo: STATe (on page 2402)	Sets or queries whether to enable the VBW for the frequency range.
[SENSe]:SPURious:RANGe<x>:DETection (on page 2403)	Sets or queries the display detector for the frequency range.
[SENSe]:SPURious:RANGe<x>:EXCursion (on page 2403)	Sets or queries the excursion level in the frequency range.
[SENSe]:SPURious:RANGe<x>:FILTer[:SHAPe] (on page 2404)	Sets or queries the filter shape for the frequency range.
[SENSe]:SPURious:RANGe<x>:FILTer[:SHAPe]: BANDwidth (on page 2404)	Sets or queries the filter bandwidth for the frequency range.
[SENSe]:SPURious:RANGe<x>:FILTer[:SHAPe]:BANDwidth:AUTO (on page 2405)	Sets or queries whether to set the filter bandwidth automatically.
[SENSe]:SPURious:RANGe<x>:FREQuency:STARt (on page 2405)	Sets or queries the start frequency of the range.

Command	Description
[SENSe]:SPURious:RANGe<x>:FREQuency:STOP (on page 2406)	Sets or queries the stop frequency of the range.
[SENSe]:SPURious:RANGe<x>:LIMit:ABSolute: STARt (on page 2406)	Sets or queries the absolute start amplitude of the limits for the range.
[SENSe]:SPURious:RANGe<x>:LIMit:ABSolute: STOP (on page 2407)	Sets or queries the absolute stop amplitude of the limits for the range.
[SENSe]:SPURious:RANGe<x>:LIMit:MASK (on page 2407)	Sets or queries the limit mask function mode for the frequency range.
[SENSe]:SPURious:RANGe<x>:LIMit:RELative: STARt (on page 2408)	Sets or queries the relative start amplitude of the limits for the range.
[SENSe]:SPURious:RANGe<x>:LIMit:RELative: STOP (on page 2409)	Sets or queries the relative stop amplitude of the limits for the range.
[SENSe]:SPURious:RANGe<x>:STATe (on page 2409)	Sets or queries whether to enable or disable the frequency range.
[SENSe]:SPURious:RANGe<x>:THReshold (on page 2410)	Sets or queries the threshold level to detect spurious signals in a range.
[SENSe]:SPURious:REFerence (on page 2410)	Sets or queries the power reference.
[SENSe]:SPURious:REFerence:MANual:POWer (on page 2411)	Sets or queries the reference power level.
[SENSe]:SQUality subgroup	**Signal quality**
[SENSe]:SQUality[:MEASview<y>]:EVM: ALL[:STATe] (on page 2411)	Sets or queries the display of Offset EVM results.
[SENSe]:SIGNalvu subgroup	**Channel trigger**
[SENSe]:SIGNalvuSVPC{{}}:ACQuisition:CONTrol: CHANnel (on page 2413)	Sets or queries the Channel Settings checkbox state.
[SENSe]:SIGNalvu:ACQuisition:CONTrol:TRIGger (on page 2412)	Sets or queries the Trigger Settings checkbox state.
[SENSe]:TOVerview subgroup	**Time overview**
[SENSe]:TOVerview[:MEASview<y>]:FREQuency: CENTer (on page 2416)	Sets or queries the center frequency.
[SENSe]:TOVerview[:MEASview<y>]: MAXTracepoints (on page 2416)	Sets or queries the maximum trace points.
[SENSe]:TOVerview[:MEASview<y>]:TIME:PER: DIVision (on page 2417)	Sets or queries the time per division (dB/div) value in the Time Overview display.
[SENSe]:USETtings subgroup	**All measurements**
[SENSe]:USETtings (on page 2425)	Updates the analyzer settings.

Command	Description
[SENSe}:WLAN subgroup	**Wireless LAN settings**
[SENSe]:WLAN:ANALysis:LENGth (on page 2442)	Specifies the length of the analysis period.
[SENSe]:WLAN:ANALysis:LENGth:ACTual? (on page 2443)	Returns the value of the actual analysis length.
[SENSe]:WLAN:ANALysis:LENGth:AUTO (on page 2443)	Sets the analysis length value based on the requirements of the selected display.
[SENSe]:WLAN:ANALysis:OFFSet (on page 2443)	Specifies or queries the analysis offset in the symbol interval.
[SENSe]:WLAN:BURSt (on page 2464)	Specifies the burst to measure when multiple bursts are present in an acquisition.
[SENSe]:WLAN:CHANnel[:BANDwidth|:BWIDth] (on page 2465)	Specifies or queries a nominal channel bandwidth to use for the WLAN measurements.
[SENSe]:WLAN:EQUalizer:TRAining (on page 2465)	Specifies the method for estimating channel frequency response and equalization.
[SENSe]:WLAN:FFT:LENGth? (on page 2466)	Returns the fixed FFT length for the WLAN measurements.
[SENSe]:WLAN:FFT:SRATe? (on page 2466)	Returns the FFT sample rate for the WLAN measurements.
[SENSe]:WLAN:FREQuency:ERRor (on page 2467)	Sets or queries the frequency error in the WLAN analysis parameters.
[SENSe]:WLAN:FREQuency:ERRor:AUTO (on page 2467)	Turns on the automatic determination of the frequency error.
[SENSe]:WLAN:GUARd:INTerval (on page 2468)	Sets or queries the Guard Interval to use in the input signal.
[SENSe]:WLAN:MODulation:DETect:QAM1024 (on page 2468)	Specifies a fixed modulation type.
[SENSe]:WLAN:MODulation:SOURce (on page 2469)	Specifies the method of data symbol modulation identification.
[SENSe]:WLAN:PILot:TRACking:AMPLitude:STATe (on page 2470)	Specifies the pilot subcarrier to use to correct amplitude variations over the packet.
[SENSe]:WLAN:PILot:TRACking:PHASe:STATe (on page 2470)	Specifies the pilot subcarrier to use to correct phase variations over the packet.
[SENSe]:WLAN:PILot:TRACking:TIMing:STATe (on page 2471)	Specifies the pilot subcarrier to use to correct timing variations over the packet.
[SENSe]:WLAN:RADix (on page 2471)	Specifies or queries the format for displaying the symbol values.
[SENSe]:WLAN:SCARriers (on page 2471)	Sets or queries which subcarriers to display in the WLAN view.
[SENSe]:WLAN:SCARriers:RANGe:STARt (on page 2472)	Sets or queries the start value of the subcarrier range.
[SENSe]:WLAN:SCARriers:RANGe:STOP (on page 2472)	Sets or queries the stop value of the subcarrier range.

Command	Description
[SENSe]:WLAN:SCARriers:SINGle:INDex (on page 2473)	Specifies or queries the single subcarrier to display in the WLAN view.
[SENSe]:WLAN:SCARriers:SPACing (on page 2473)	Sets or queries the subcarrier spacing.
[SENSe]:WLAN:SCARriers:SPACing:AUTO (on page 2474)	Automatically sets the subcarrier spacing.
[SENSe]:WLAN:SIGNal:STANdard:TYPE (on page 2474)	Sets or queries the WLAN signal standard.
[SENSe]:WLAN:SubCARrier:DERotation (on page 2475)	Shows subcarriers for displays with or without the Gamma subcarrier phase rotation removed.
[SENSe]:WLAN:SUMMary:CLEar:RESults (on page 2475)	Clears the results of the WLAN Summary display.
[SENSe]:WLAN:SUMMary:CLEar:RESults (on page 2475)	Clears the results in the WLAN Summary measurement.
[SENSe]:WLAN:SUMMary:EVM:RMS:AVERage: COUNt (on page 2476)	Specifies the maximum number of bursts to average.
[SENSe]:WLAN:SUMMary:EVM:RMS:AVERage: COUNt:ENABle (on page 2476)	Enables the burst average function.
[SENSe]:WLAN:SWAP:IQ (on page 2477)	Specifies to swap the I and Q components of a signal.
[SENSe]:WLAN:SYMBols|CHIPs (on page 2477)	Specifies or queries the symbols or chips are displayed in the WLAN Symbol Table.
[SENSe]:WLAN:SYMBols|CHIPs:MAX (on page 2478)	Sets or queries the value for the maximum number of symbols/chips to analyze.
[SENSe]:WLAN:SYMBols|CHIPs:MAX:STATe (on page 2478)	Turns the maximum data symbols to analyze feature on or off.
[SENSe]:WLAN:SYMBols|CHIPs:RANGe:COUNt (on page 2479)	Sets or queries the number of symbols or chips to display.
[SENSe]:WLAN:SYMBols|CHIPs:RANGe:STARt (on page 2479)	Sets or queries the symbols/chips start value.
[SENSe]:WLAN:SYMBols|CHIPs:RANGe:STOP (on page 2480)	Sets or queries the symbols or chips stop value.
[SENSe]:WLAN:SYMBols|CHIPs:SINGle:INDex (on page 2480)	Sets or queries the symbol or chip to use when displaying results from a single symbol.
[SENSe]:WLAN:UNIT:FREQuency (on page 2480)	Specifies or queries whether the displayed frequency units are frequency (Hz) or subcarrier.
[SENSe]:WLAN:UNIT:TIME (on page 2481)	Specifies or queries whether the displayed time units are seconds, symbols or chips.
SENSe:LTE subgroup	
SENSe:LTE:ACLR:BANDwidth:RESolution (on page 2148)	Sets or queries the resolution bandwidth (RBW).

Command	Description
SENSe:LTE:ACLR:BANDwidth:VIDeo (on page 2148)	Sets or queries the video bandwidth (VBW).
SENSe:LTE:ACLR:BANDwidth:VIDeo:STATe (on page 2148)	Sets or queries enabling or disabling the video bandwidth (VBW).
SENSe:LTE:ACLR:CHANnel:IBANdwidth (on page 2149)	Sets or queries the reference channel integration bandwidth.
SENSe:LTE:ACLR:CHANnel:WIDTh (on page 2149)	Sets or queries the reference channel width.
SENSe:LTE:ACLR:CLEar:RESults (on page 2150)	Clears the results in the measurement.
SENSe:LTE:ACLR:FREQuency (on page 2150)	Sets or queries the measurement center frequency.
SENSe:LTE:ACLR:MFILter (on page 2151)	Sets or queries the shape of the filter.
SENSe:LTE:ACLR:MODE:REALtime (on page 2151)	Sets or queries the real-time LTE ACLR measurement mode.
SENSe:LTE:ACLR:NFLoor:ACQuire (on page 2152)	Enables measurement of the noise floor.
SENSe:LTE:ACLR:NFLoor:CORRected? (on page 2152)	Returns whether or not noise correction was applied on the last acquisition.
SENSe:LTE:ACLR:NFLoor:STATe (on page 2153)	Determines whether to enable or disable the correction for noise floor.
SENSe:LTE:ACLR:OFFSet<x>:BANDwidth: RESolution (on page 2153)	Sets or queries the resolution bandwidth (RBW) value in the specified row in the Offsets & Limits table.
SENSe:LTE:ACLR:OFFSet<x>:BANDwidth: RESolution:COUNt (on page 2154)	Sets or queries the count for the resolution bandwidths in the specified row in the Offsets & Limits table.
SENSe:LTE:ACLR:OFFSet<X>:BANDwidth: RESolution:FILTer (on page 2154)	Sets or queries the filter bandwidth resolution under Offset & Limit table.
SENSe:LTE:ACLR:OFFSet<x>:BANDwidth: RESolution:VIDeo (on page 2155)	Sets or queries the resolution video bandwidth (VBW) value in the specified row in the Offsets & Limits table.
SENSe:LTE:ACLR:OFFSet<x>:BANDwidth: RESolution:VIDeo:STATe (on page 2155)	Sets or queries whether to enable or disable the video resolution bandwidth (VBW) in the specified row in the Offsets & Limits table.
SENSe:LTE:ACLR:OFFSet<x>:DETection (on page 2156)	Sets or queries the detection settings for all offsets.
SENSe:LTE:ACLR:OFFSet<x>:FREQuency: BANDwidth (on page 2156)	Sets or queries the frequency bandwidth value for the specified row of the Offsets & Limits table.
SENSe:LTE:ACLR:OFFSet<x>:FREQuency:OFFSet (on page 2157)	Sets or queries the frequency offset value of the specified row in the Offsets & Limits table.
SENSe:LTE:ACLR:OFFSet<x>:LIMit:ABSolute (on page 2157)	Sets or queries the absolute limit value of the row in the Offset & Limits table.
SENSe:LTE:ACLR:OFFSet<x>:LIMit:MASK (on page 2158)	Sets or queries the limit mask setting for the specified row in the Offsets & Limits table.

Command	Description
SENSe:LTE:ACLR:OFFSet<x>:LIMit:RELative (on page 2158)	Sets or queries the relative limit value of channel under Offset & Limits table.
SENSe:LTE:ACLR:OFFSet<x>:LIMit:SIDE (on page 2159)	Sets or queries the side of the offset limit for the specified row in the Offsets & Limits table.
SENSe:LTE:ACLR:OFFSet<x>:STATe (on page 2159)	Sets or queries the specified row of the Offsets & Limits table.
SENSe:LTE:ACLR:POINts:COUNt (on page 2160)	Sets or queries the number of points used for the trace.
SENSe:LTE:ACLR:REFerence:MAGNitude:AUTO (on page 2161)	Sets or queries the reference magnitude to automatic or manual.
SENSe:LTE:ACLR:REFerence:MAGNitude: DETection (on page 2161)	Sets or queries the reference magnitude detection.
SENSe:LTE:ACLR:REFerence:MAGNitude (on page 2160)	Sets or queries the reference power magnitude.
SENSe:LTE:ACLR:STEP:AUTO (on page 2162)	Determines whether to automatically update the center frequency increment/decrement step size.
SENSe:LTE:ACLR:STEP:MAGNitude (on page 2162)	Sets or queries the step magnitude for the center frequency.
SENSe]:LTE:ANALysis:EQUAlization:STATe (on page 2163)	Sets or queries the Enable Equalization state in the LTE Analysis Params tab.
SENSe:LTE:ANALysis:LENGth:ACTual? (on page 2164)	Queries the actual analysis length for the LTE Constellation and LTE Power vs Time displays.
[SENSe]:LTE:ANALysis:OFFSet (on page 2165)	Sets or queries the LTE analysis offset for the LTE Constellation or LTE Power vs Time.
SENSe:LTE:ANALysis:OFFSet:AUTO (on page 2166)	Sets or queries the Auto checkbox state for analysis offset.
SENSe:LTE:ANALysis:TIMe:UNIT (on page 2166)	Sets or queries the unit preference under the LTE Analysis tab.
SENSe:LTE:CHANnel:BANDwidth (on page 2169)	Specifies or queries the channel bandwidth to use.
SENSe:LTE:CHSPectrum:AVERage (on page 2169)	Sets or queries how to average the waveform.
SENSe:LTE:CHSPectrum:AVERage:COUNt (on page 2170)	Sets or queries the number of traces for averaging.
SENSe:LTE:CHSPectrum:BANDwidth:RESolution (on page 2170)	Sets or queries the resolution bandwidth (RBW).
SENSe:LTE:CHSPectrum:BANDwidth:RESOlution: AUTO (on page 2171)	Determines whether to set the resolution bandwidth (RBW) automatically or manually. Queries this setting.
SENSe:LTE:CHSPectrum:BANDwidth:VIDeo (on page 2171)	Sets or queries the video bandwidth (VBW).
SENSe:LTE:CHSPectrum:BANDwidth:VIDeo:STATe (on page 2172)	Sets or queries enabling or disabling the video bandwidth (VBW).

Command	Description
SENSe:LTE:CHSPectrum:CHANnel:BANDwidth (on page 2172)	Specifies or queries a nominal channel bandwidth to use.
SENSe:LTE:CHSPectrum:CLEar:RESults (on page 2172)	Clears the results in the LTE Channel Spectrum measurement.
SENSe:LTE:CHSPectrum:DETection (on page 2173)	Sets or queries the trace detection settings.
SENSe:LTE:CHSPectrum:FREQuency (on page 2173)	Sets or queries the measurement center frequency.
SENSe:LTE:CHSPectrum:FREQuency:SPAN (on page 2174)	Sets or queries the frequency span.
SENSe:LTE:CHSPectrum:FREQuency:STEP (on page 2174)	Sets or queries the frequency step size.
SENSe:LTE:CHSPectrum:FREQuency:STEP:AUTO (on page 2175)	Determines whether to set the frequency step size automatically or manually. Queries this setting.
SENSe:LTE:CHSPectrum:POINts:COUNt (on page 2175)	Sets or queries the number of points used for the trace.
SENSe:LTE:FRAMe:STRUcture (on page 2176)	Sets or queries the frame structure.
[SENSe]: TDIagram	**TDIagram measurement**
[SENSe]:TDIagram[:MEASview<y>]:SELEcted: SOURce (on page 2415)	Sets or queries the source of Trellis diagram for a selected Measurement view number.
SENSe:TOFF:POWEr:SPECtral:DENSity (on page 2415)	Sets or queries the limit set for comparison of the Toff power spectral density value.

Status commands

Use the STATus commands to control registers defined in the SCPI status reporting structure. See the Index for commands not listed in the following table.

Status commands

Header	Description
STATus:ACPower:EVENts? (on page 2483)	Returns the current events for the ACPR measurement.
STATus:BIBEmissions:EVENts? (on page 2486)	Returns the current events and status conditions for the Bluetooth InBand Emission measurement.
STATus:BLUEtooth:CONSte:EVENts? (on page 2486)	Returns the current events and status conditions for the Bluetooth Constellation measurement.
STATus:BLUEtooth:EDIAgram:EVENts? (on page 2487)	Returns the current events and status conditions for the Bluetooth Eye Diagram measurement.
STATus:BLUEtooth:FDVTime:EVENts? (on page 2487)	Returns the current events and status conditions for the Frequency Deviation versus Time measurement.

Header	Description
STATus:P25:EDIagram:EVENts? (on page 2505)	Returns the current events and status conditions for the P25 Eye Diagram measurement.
STATus:{AM|FM|PM}:EVENts? (on page 2483)	Returns the current events for the AM/FM/PM measurement.
STATus:AUDio:SPECtrum:EVENts? (on page 2484)	Returns the current events and status conditions for the audio measurement.
STATus:AUDio:SUMMary:EVENts? (on page 2484)	Returns the current events and status conditions for the audio measurement.
STATus:AVTime[:MEASview<y>]:EVENts? (on page 2485)	Returns the current events for the Amplitude versus Time measurement.
STATus:CCDF:EVENts? (on page 2488)	Returns the current events for the CCDF measurement.
STATus:CONSte[:MEASview<x>]:EVENts? (on page 2489)	Returns the current events and status conditions for the constellation measurement.
STATus:DIQVtime[:MEASview<y>]:EVENts? (on page 2490)	Returns the current events and status conditions for the Demod I&Q versus Time measurement.
STATus:DPX:EVENts? (on page 2490)	Returns the current events for the DPX spectrum measurement.
STATus:DPX:EVENts?	Returns the current events for the DPX spectrum measurement.
STATus:EDIagram[:MEASview<y>]:EVENts? (on page 2491)	Returns the current events and status conditions for the eye diagram measurement.
STATus:EVM[:MEASview<y>]:EVENts? (on page 2492)	Returns the current events and status conditions for the EVM versus Time measurement.
STATus:FDVTime[:MEASview<y>]:EVENts? (on page 2493)	Returns the current events and status conditions for the Frequency deviation versus Time measurement.
STATus:{FSETtling|PSETtling}:EVENts? (on page 2493)	Returns the current events for the specified settling time measurement.
STATus:FVTime[:MEASview<y>]:EVENts? (on page 2494)	Returns the current events for the Frequency versus Time measurement.
STATus:IQVTime[:MEASview<y>]:EVENts? (on page 2495)	Returns the current events for the RF I&Q versus Time measurement.
STATus:MCPower:EVENts? (on page 2497)	Returns the current events for the MCPR measurement.
STATus:MERRor[:MEASview<y>]:EVENts? (on page 2498)	Returns the current events and status conditions for the Magnitude error versus Time measurement.
STATus:OBWidth:EVENts? (on page 2498)	Returns the current events for the Occupied Bandwidth measurement.
STATus:OFDM:CONSte:EVENts? (on page 2499)	Returns the current events and status condition for the OFDM Constellation measurement.

Header	Description
STATus:OFDM:EVM:EVENts? (on page 2500)	Returns the current events and status condition for the OFDM EVM (Error Vector Magnitude) versus Time measurement.
STATus:OFDM:MERRor:EVENts? (on page 2500)	Returns the current events and status condition for the OFDM Magnitude versus Time measurement.
STATus:OFDM:PERRor:EVENts? (on page 2501)	Returns the current events and status condition for the OFDM Phase error versus Time measurement.
STATus:OFDM:POWer:EVENts? (on page 2501)	Returns the current events and status condition for the OFDM Power measurement.
STATus:OFDM:STABle:EVENts? (on page 2502)	Returns the current events and status condition for the OFDM Symbol table measurement.
STATus:OPERation:CONDition? (on page 2503)	Queries the contents of the OCR.
STATus:OPERation:ENABle (on page 2503)	Sets or queries the mask for the OENR.
STATus:OPERation[:EVENt]? (on page 2504)	Queries the contents of the OEVR.
STATus:OPERation:NTRansition (on page 2504)	Sets or queries the value of the negative transition filter.
STATus:OPERation:PTRansition (on page 2505)	Sets or queries the value of the positive transition filter.
STATus:PERRor[:MEASview<y>]:EVENts? (on page 2506)	Returns the current events and status conditions for the Phase error versus Time measurement.
STATus:PHVTime[:MEASview<y>]:EVENts? (on page 2507)	Returns the current events for the Phase versus Time measurement.
STATus:PNOise:EVENts? (on page 2507)	Returns the current events for the phase noise measurement.
STATus:PRESet (on page 2508)	Presets a status byte.
STATus:PULSe:RESult[:MEASview<y>]:EVENts? (on page 2510)	Returns the current events for the pulse table measurement.
STATus:PULSe:STATistics[:MEASview<y>]:EVENts? (on page 2511)	Returns the current events for the pulse statistics measurement.
STATus:PULSe:TRACe[:MEASview<y>]:EVENts? (on page 2511)	Returns the current events for the pulse trace measurement.
STATus:QUEStionable:CONDition? (on page 2514)	Queries the contents of the QCR.
STATus:QUEStionable:ENABle (on page 2515)	Sets or queries the mask for the OENR.
STATus:QUEStionable[:EVENt]? (on page 2515)	Queries the contents of the QER.
STATus:QUEStionable:NTRansition (on page 2518)	Sets or queries the value of the negative transition filter.

Header	Description
STATus:QUEStionable:PTRansition (on page 2519)	Sets or queries the value of the positive transition filter.
STATus:QUEStionable:CALibration:CONDition? (on page 2512)	Queries the contents of the questionable calibration condition register.
STATus:QUEStionable:CALibration:ENABle (on page 2512)	Sets or queries the mask for the questionable calibration enable register.
STATus:QUEStionable:CALibration[:EVENt]? (on page 2513)	Queries the contents of the questionable calibration event register.
STATus:QUEStionable:CALibration:NTRansition (on page 2513)	Sets or queries the value of the negative transition filter.
STATus:QUEStionable:CALibration:PTRansition (on page 2514)	Sets or queries the value of the positive transition filter.
STATus:QUEStionable:FREQuency:CONDition? (on page 2516)	Queries the contents of the questionable frequency condition register.
STATus:QUEStionable:FREQuency:ENABle (on page 2516)	Sets or queries the mask for the questionable frequency enable register.
STATus:QUEStionable:FREQuency[:EVENt]? (on page 2517)	Queries the contents of the questionable frequency event register.
STATus:QUEStionable:FREQuency:NTRansition (on page 2517)	Sets or queries the value of the negative transition filter.
STATus:QUEStionable:FREQuency:PTRansition (on page 2518)	Sets or queries the value of the positive transition filter.
STATus:SEM:EVENts? (on page 2519)	Returns the current events and status conditions for the Spectral Emissions Mask measurement.
STATus:SGRam[:MEASview<y>]:EVENts? (on page 2520)	Returns the current events for the spectrogram measurement.
STATus:SPECtrum[:MEASview<y>]:EVENts? (on page 2521)	Returns the current events for the spectrum measurement.
STATus:SPURious:EVENts? (on page 2521)	Returns the current events for the spurious measurement.
STATus:SQUality[:MEASview<y>]:EVENts? (on page 2522)	Returns the current events and status conditions for the signal quality measurement.
STATus:TDIagram[:MEASview<y>]:EVENts? (on page 2522)	Returns the current events and status conditions for the trellis diagram measurement.
STATus:WLAN:CONSte:EVENts? (on page 2523)	Returns the current status concatenated with WLAN Constellation-specific event information.

Header	Description
STATus:WLAN:CRESponse:EVENts? (on page 2525)	Returns the current events and status conditions for the WLAN Channel Response display.
STATus:WLAN:EVM:EVENts? (on page 2526)	Returns the current status concatenated with WLAN EVM-specific event information.
STATus:WLAN:MERRor:EVENts? (on page 2528)	Returns the current status concatenated with WLAN Magnitude Error-specific event information.
STATus:WLAN:PERRor:EVENts? (on page 2529)	Returns status concatenated with WLAN Phase Error-specific event information.
STATus:WLAN:PVTime:EVENts? (on page 2530)	Returns the current status concatenated with WLAN Power vs. Time-specific event information.
STATus:WLAN:STABle:EVENts? (on page 2530)	Returns status concatenated with WLAN Symbol Table-specific event information.
STATus:WLAN:SUMMary:EVENts? (on page 2531)	Returns the current events and status conditions for the WLAN summary measurement.
STATus:TXGain subgroup	**Transmission Gain measurements**
STATus:TXGain:EVENts? (on page 2523)	Queries events for the Transmission Gain display.
STATus:LTE subgroup	**LTE analysis**
STATus:LTE:ACLR:EVENts? (on page 2495)	Returns the current events and status conditions for this display.
STATus:LTE:CHSPectrum:EVENts? (on page 2496)	Returns the current events and status conditions for this display.
STATus:LTE:CONSte:EVENts? (on page 2496)	Returns the current events and status conditions for this display.
STATus:LTE:PVTime:EVENts? (on page 2497)	Returns the current events and status conditions for this display.

Source commands

Use the SOURce commands to control output power level of the tracking generator. See the Index for commands not listed in the following table.

These commands only apply when a RSA500A and RSA600A Series instrument with Option 04 (Tracking Generator) installed is connected.

Source commands

Command	Description
SOURce:TXGain:POWer (on page 2481)	Sets or queries the output power level of the tracking generator.

System commands

Use the SYSTem commands to set or query system parameters for operation. See the Index for commands not listed in the following table.

System commands

Header	Description
SYSTem:COMMunicate:GPIB[:SELF]:ADDRess (on page 2541)	Sets or queries the GPIB address of the instrument.
SYSTem:PRESet:BLUEtooth:STANdard (on page 2552)	Sets or queries the standard, setup, Retaining Current Reflevel, and Retaining Center Frequency in the Bluetooth standard preset.
SYSTem:PRESet:P25:STANdard (on page 2556)	Restores the analyzer defaults for the P25 Standards preset.
SYSTem:DATE (on page 2541)	Sets or queries the current date.
SYSTem:ERRor:ALL? (on page 2542)	Queries all the error or event information.
SYSTem:ERRor:CODE:ALL? (on page 2542)	Queries all the error or event codes.
SYSTem:ERRor:CODE[:NEXT]? (on page 2543)	Queries the latest error or event information.
SYSTem:ERRor:COUNt? (on page 2543)	Queries the number of errors or events.
SYSTem:ERRor[:NEXT]? (on page 2543)	Queries the latest error or event information.
SYSTem:KLOCk (on page 2550)	Enables or disables the local lockout operation.
SYSTem:LINK:SETTings (on page 2550)	Sets or queries whether or not global settings linking enabled for the measurements displayed.
SYSTem:LINK:SETTings:SYNC (on page 2550)	Sets the synchronize settings referring to source for the measurements displayed.
SYSTem:OPTions? (on page 2551)	Queries the optional information.
SYSTem:PRESet (on page 2551)	Restores the analyzer to the defaults.
SYSTem:PRESet:APPLication (on page)	Restores the analyzer to the defaults for the application preset type.
SYSTem:PRESet:APPLication:ACTion (on page 2552)	Sets or queries the preset action for the application preset type.
SYSTem:PRESet:APPLication:SELected (on page)	Sets or queries the presets for the application preset type.
SYSTem:PRESet:DPX (on page 2553)	Presets the analyzer.
SYSTem:PRESet:DPX:ACTion (on page 2553)	Sets or queries the preset action for the DPX preset type.
SYSTem:PRESet:DPX:SELected (on page 2553)	Sets or queries the presets for the DPX preset type.
SYSTem:PRESet[:MAIN] (on page 2555)	Restores the analyzer to the defaults.
SYSTem:PRESet:MAIN:ACTion (on page 2555)	Sets or queries the preset action for the main preset type.

Header	Description
SYSTem:PRESet:MAIN:SELected (on page 2555)	Sets or queries the presets for the main preset type.
SYSTem:PRESet:STANdards (on page 2556)	Restores the instrument to the defaults for the WLAN standards preset type.
SYSTem:PRESet:STANdards:ACTion (on page 2557)	Sets or queries the preset action for the standards preset type.
SYSTem:PRESet:USER (on page 2557)	Restores the analyzer to the defaults for the user preset type.
SYSTem:PRESet:USER:ACTion (on page 2558)	Sets or queries the preset action for the user preset type.
SYSTem:PRESet:USER:SELected (on page 2558)	Sets or queries the presets for the user preset type.
SYSTem:PRESet:WLAN:STANdard (on page 2560)	Sets or queries the standard and bandwidth presets for the WLAN standards preset type.
SYSTem:TIME (on page 2565)	Sets or queries the current time.
SYSTem:VERSion? (on page 2566)	Queries the version of the SCPI.
SYSTem:PRESet:LTE:STANdard (on page 2554)	Sets or queries LTE standard preset parameters.
SYSTem:SVPC:SOURce (on page 2563)<x> (on page 2564):ADD (on page 2563)	Adds Measurement Source.
SYSTem:SVPC:SOURce (on page 2563)<x> (on page 2564):REMOve (on page 2563)	Removes latest user defined measurement Source from source configuration
SYSTem:SVPC:SOURce (on page 2564)<x> (on page 2564):REMOve:ALL (on page 2564)	Removes all user defined measurement Sources from source configuration
SYSTem:SVPC:SOURce<x>:RENAme (on page 2564)	Sets or queries measurement Source name under source configuration

Trace commands

Use the TRACe commands to select trace type and to control trace arithmetic. See the Index for commands not listed in the following table.

Trace commands

Commands	Description
TRACe<x>:{AM|FM|PM} subgroup	**AM/FM/PM measurement**
TRACe<x>:{AM|FM|PM} (on page 2620)	Determines whether or not to show the specified trace.
TRACe:{AM|FM|PM}:DETection (on page 2566)	Sets or queries the display detector, the method to be used for decimating traces to fit the available horizontal space on screen.
TRACe:{AM|FM|PM}:FREeze (on page 2567)	Sets or queries whether or not to freeze the trace display.

Commands	Description
TRACe:{AM|FM|PM}:FUNCtion (on page 2567)	Sets or queries the trace function.
TRACe<x>:Bluetooth subgroup	**Bluetooth measurement**
TRACe<x>:BLUEtooth:CONSte (on page 2631)	Determines whether or not to show the specified trace.
TRACe<x>:BLUEtooth:CONSte:FREeze (on page 2632)	Sets or queries whether or not to freeze the specified trace.
TRACe<x>:BLUEtooth:CONSte:MODE (on page 2632)	Sets or queries the trace display mode.
TRACe<x>:BLUEtooth:CONSte:SELect (on page 2633)	Selects the specified trace.
TRACe<x>:BLUEtooth:EDIagram:ENABle:I (on page 2633)	Determines whether to show or hide the I trace.
TRACe<x>:BLUEtooth:EDIagram:ENABle:Q (on page 2633)	Determines whether to show or hide the Q trace.
TRACe<x>:BLUEtooth:EDIagram:SELect:I (on page 2634)	Selects the I trace or queries if the I trace is selected or not.
TRACe<x>:BLUEtooth:EDIagram:SELect:Q (on page 2634)	Selects the Q trace or queries if the Q trace is selected or not.
TRACe<x>:BLUEtooth:FDVTime:MODE (on page 2635)	Sets or queries whether to display the Frequency Deviation vs. Time trace as vectors or symbols (points).
TRACe:P25 subgroup	**APCO P25 measurement**
TRACe<x>:P25:CONStellation (on page 2678)	Determines whether or not to show the specified trace.
TRACe<x>:P25:CONStellation:FREeze (on page 2678)	Determines whether or not to freeze (halt updates of) the specified trace.
TRACe<x>:P25:CONStellation:MODE (on page 2679)	Sets or queries how to display the trace content.
TRACe<x>:P25:CONStellation:SELect (on page 2679)	Selects the specified trace.
TRACe<x>:P25:EDIagram:ENABle:I (on page 2680)	Determines whether to show or hide the I trace.
TRACe<x>:P25:EDIagram:ENABle:Q (on page 2680)	Determines whether to show or hide the Q trace.
TRACe<x>:P25:EDIagram:SELect:I (on page 2681)	Selects the I trace or queries if the I trace is selected or not.
TRACe<x>:P25:EDIagram:SELect:Q (on page 2682)	Selects the Q trace or queries if the Q trace is selected or not.
TRACe:Noise subgroup	**Noise Figure and Gain measurement**
TRACe<x>:NOISe:FIGure:AVERageCOUNt (on page 2656)	Sets or queries the average counts for the specified trace.
TRACe<x>:NOISe:FIGure:COUNt:ENABle (on page 2657)	Sets to enable the count feature, or queries the enables status, of the specified trace.
TRACe<x>:NOISe:FIGure:DETection (on page 2657)	Sets or queries the trace detection method.

Commands	Description
TRACe<x>:NOISe:FIGure:FREeze (on page 2658)	Determines whether or not to freeze (halt updates of) the specified trace.
TRACe<x>:NOISe:FIgure:FUNCtion (on page 2658)	Sets or queries the trace processing method (function) in the display.
TRACe<x>:NOISe:FIGure:SELect (on page 2659)	Selects the specified trace.
TRACe<x>:NOISe:FIGure:SHOW (on page 2659)	Determines whether or not to show the specified trace.
TRACe<x>:NOISe:GAIN:AVERage:COUNt (on page 2660)	Sets or queries the average counts for the specified trace.
TRACe<x>:NOISe:GAIN:COUNt:ENABle (on page 2661)	Enables or disables the count feature for the specified trace.
TRACe<x>:NOISe:GAIN:DETection (on page 2661)	Sets or queries the trace detection method.
TRACe<x>:NOISe:GAIN:FREeze (on page 2662)	Determines whether or not to freeze (halt updates of) the specified trace.
TRACe<x>:NOISe:GAIN:FUNCtion (on page 2662)	Sets or queries the trace processing method (function) in the display.
TRACe<x>:NOISe:GAIN:SELect (on page 2663)	Selects the specified trace.
TRACe<x>:NOISe:GAIN:SHOW (on page 2664)	Determines whether or not to show the specified trace.
TRACe<x>:NOISe:TEMPerature:AVERageCOUNt (on page 2664)	Sets or queries the average counts for the specified trace.
TRACe<x>:NOISe:TEMPerature:COUNt:ENABle (on page 2665)	Enables or disables the count feature for the specified trace.
TRACe<x>:NOISe:TEMPerature:DETection (on page 2665)	Sets or queries the trace detection method.
TRACe<x>:NOISE:TEMPerature:FREeze (on page 2666)	Determines whether or not to freeze (halt updates of) the specified trace.
TRACe<x>:NOISe:TEMPerature:FUNCtion (on page 2666)	Sets or queries the trace processing method (function) in the display.
TRACe<x>:NOISe:TEMPerature:SELect (on page 2667)	Selects the specified trace.
TRACe<x>:NOISe:TEMPerature:SHOW (on page 2667)	Determines whether or not to show the specified trace.
TRACe<x>:NOISe:YFACtor:AVERage:COUNt (on page 2668)	Sets or queries the average counts for the specified trace.
TRACe<x>:NOISe:YFACtor:COUNt:ENABle (on page 2669)	Enables or disables the count feature for the specified trace.
TRACe<x>:NOISe:YFACtor:DETection (on page 2669)	Sets or queries the trace detection method.
TRACe<x>:NOISE:YFACtor:FREeze (on page 2670)	Determines whether or not to freeze (halt updates of) the specified trace.

Commands	Description
TRACe<x>:NOISe:YFACtor:FUNCtion (on page 2670)	Sets or queries the trace processing method (function) in the display.
TRACe<x>:NOISe:YFACtor:SELect (on page 2671)	Selects the specified trace.
TRACe<x>:NOISe:YFACtor:SHOW (on page 2671)	Determines whether or not to show the specified trace.
TRACe<x>:AVTime subgroup	**Amplitude versus time measurement**
TRACe<x>:AVTime[:MEASview<y>] (on page 2621)	Sets or queries whether or not to show the specified trace.
TRACe:{FSETtling|PSETtling}:AVERage:COUNt (on page 2579)	Sets or queries the number of traces to combine for averaging.
TRACe<x>:AVTime[:MEASview<y>]:AVERage: COUNt (on page 2621)	Sets or queries the number of traces to combine.
TRACe<x>:AVTime[:MEASview<y>]:AVERage: RESet (on page 2622)	Clears the average data and resets the average counter.
TRACe<x>:AVTime[:MEASview<y>]:COLor:CURVe (on page 2622)	Sets or queries how colors are mapped to the signal density in the Amplitude versus Time bitmap display.
TRACe<x>:AVTime[:MEASview<y>]:COLor: SCALe:AUTO (on page 2623)	Automatically adjusts the Max and Min color settings to display the broadest range of colors in the Amplitude versus Time bitmap display.
TRACe<x>:AVTime[:MEASview<y>]:COUNt (on page 2623)	Sets or queries the count for the Max or Min Hold trace.
TRACe<x>:AVTime[:MEASview<y>]:COUNt:ENABle (on page 2624)	Sets or queries whether or not to enable the count for Max/Min Hold.
TRACe<x>:AVTime[:MEASview<y>]:COUNt:RESet (on page 2624)	Clears the Max or Min Hold data and counter, and restarts the process.
TRACe<x>:AVTime[:MEASview<y>]:DETection (on page 2625)	Sets or queries the display detector.
TRACe<x>:AVTime[:MEASview<y>]:DOT:PERSistent (on page 2626)	Determines whether to enable or disable the dot persistence for the bitmap trace (Trace 5) in the Amplitude versus Time view.
TRACe<x>:AVTime[:MEASview<y>]:DOT:PERSistent: TYPE (on page 2626)	Sets or queries the persistence type for the bitmap trace (Trace 5) in the Amplitude versus Time view.
TRACe<x>:AVTime[:MEASview<y>]:DOT:PERSistent: VARiable (on page 2627)	Sets or queries how long data points are displayed.
TRACe<x>:AVTime[:MEASview<y>]:FREeze (on page 2627)	Sets or queries whether or not to freeze the display of the trace.
TRACe<x>:AVTime[:MEASview<y>]:FUNCtion (on page 2628)	Sets or queries the trace function.
TRACe<x>:AVTime[:MEASview<y>]:LEFToperand (on page 2628)	Sets or queries the left operand for the math trace.

Commands	Description
TRACe<x>:AVTime[:MEASview<y>]:RIGHtoperand (on page 2629)	Sets or queries the right operand for the math trace.
TRACe<x>:AVTime[:MEASview<y>]:SELect (on page 2630)	Sets or queries the trace number to display the readout.
TRACe<x>:AVTime[:MEASview<y>]:SELect:LINe (on page 2630)	Sets or queries the line number to send to the Ampliude vs time ogram display.
TRACe<x>:CCDF subgroup	**CCDF measurement**
TRACe<x>:CCDF:FREeze (on page 2635)	Sets or queries whether or not to freeze the display of the trace.
TRACe<x>:CCDF:SELect (on page 2636)	Sets or queries the trace number to display the readout.
TRACe<x>:CCDF:SHOW (on page 2636)	Sets or queries whether to show or hide the trace.
TRACe<x>:CCDF:X (on page 2637)	Sets or queries the horizontal position of the measurement pointer.
TRACe<x>:CCDF:Y? (on page 2637)	Queries the vertical position (CCDF value) of the measurement pointer.
TRACe:CONSte subgroup (Option 21 only)	**Constellation measurement**
TRACe<x>:CONSte[:MEASview<y>] (on page 2638)	Specifies whether or not to show the specified trace (Trace 1 or 2) in the Constellation display.
TRACe<x>:CONSte[:MEASview<y>]:FREeze (on page 2639)	Halts updates to the specified trace in the Constellation display.
TRACe<x>:CONSte[:MEASview<y>]:MODE (on page 2639)	Sets or queries how to display the constellation trace.
TRACe<x>:CONSte[:MEASview<y>]:Q:OFFSet (on page 2640)	Sets the Q offset in the Constellation display to Use shared Pref, remove Q offset, or include Q offset.
TRACe<x>:CONSte[:MEASview<y>]:SELect (on page 2640)	Sets or queries the trace to which the Show, Freeze, Content, and Q Offset settings apply.
TRACe:DIQVtime subgroup (Option 21 only)	**Demodulated I&Q versus time measurement**
TRACe:DIQVtime[:MEASview<y>]:ENABle:I (on page 2567)	Enable or disables the trace I.
TRACe:DIQVtime[:MEASview<y>]:ENABle:Q (on page 2568)	Enable or disables the Q trace.
TRACe:DIQVtime[:MEASview<y>]:MODE (on page 2568)	Sets or queries whether to display the Demod I&Q vs Time trace as vectors or symbols.
TRACe:DIQVtime[:MEASview<y>]:SELect:I (on page 2569)	Selects the I trace.
TRACe:DIQVtime[:MEASview<y>]:SELect:Q (on page 2570)	Selects the Q trace.
TRACe<x>:DPX subgroup	**DPX spectrum measurement**
TRACe<x>:DPX (on page 2641)	Sets or queries whether or not to show the waveform.

Commands	Description
TRACe<x>:DPX:AVERage:COUNt (on page 2642)	Sets or queries the number of traces to combine for averaging.
TRACe<x>:DPX:COLor:CURVe (on page 2642)	Sets or queries the color intensity.
TRACe<x>:DPX:COLor:INTensity (on page 2642)	Sets or queries the color intensity in the DPX spectrum view.
TRACe<x>:DPX:COLor:SCALe:AUTO (on page 2643)	Automatically adjusts the Max and Min color settings to display the broadest range of colors.
TRACe<x>:DPX:DETection (on page 2643)	Sets or queries the function.
TRACe<x>:DPX:DGRAM:SELect:LINE (on page 2644)	Sets or queries a line in the DPXogram display to send to the DPX spectrum display.
TRACe<x>:DPX:DOT:PERSistent (on page 2644)	Sets or queries whether to enable or disable the dot persistence.
TRACe<x>:DPX:DOT:PERSistent:TYPE (on page 2645)	Sets or queries the persistence type.
TRACe<x>:DPX:DOT:PERSistent:VARiable (on page 2645)	Sets or queries the length of time that data points are displayed.
TRACe<x>:DPX:FREeze (on page 2646)	Sets or queries whether or not to freeze the display of the trace.
TRACe<x>:DPX:FUNCtion (on page 2646)	Sets or queries the trace function for the in the DPX spectrum view.
TRACe<x>:DPX:LEFToperand (on page 2647)	Sets or queries the left operand for the math trace.
TRACe<x>:DPX:RIGHtoperand (on page 2647)	Sets or queries the right operand for the math trace.
TRACe<x>:DPX:SELect (on page 2648)	Sets or queries the trace number to display the readout.
TRACe:EDIagram subgroup (Option 21 only)	**Eye diagram**
TRACe:EDIagram[:MEASview<y>]:ENABle:I (on page 2570)	Determines whether to show or hide the I trace in the eye diagram.
TRACe:EDIagram[:MEASview<y>]:ENABle:Q (on page 2571)	Determines whether to show or hide the Q trace in the eye diagram.
TRACe:EDIagram[:MEASview<y>]:SELect:I (on page 2571)	Selects the I trace in the eye diagram. The query version of this command returns whether the I trace is selected or not.
TRACe:EDIagram[:MEASview<y>]:SELect:Q (on page 2572)	Selects the Q trace in the eye diagram. The query version of this command returns whether the Q trace is selected or not.
TRACe:EVM subgroup	**EVM measurement**
TRACe:EVM[:MEASview<y>]:MODE (on page 2578)	Sets or queries whether to display the EVM versus Time trace as vectors or symbols (points).
TRACe:FDVTime subgroup	**Frequency deviation versus time measurement**

Commands	Description
TRACe:FDVTime:MODE (on page 2578)	Sets or queries whether to display the Frequency Deviation versus Time trace as vectors or symbols.
TRACe:FVTime subgroup	**Frequency versus time measurement**
TRACe:FVTime[:MEASview<y>] (on page 2582)	Sets or queries whether or not to show the trace.
TRACe:FVTime[:MEASview<y>]:AVERage:COUNt (on page 2582)	Sets or queries the number of traces to combine for averaging.
TRACe<x>:FVTime[:MEASview<y>]:COLor:CURVe (on page 2583)	Sets or queries how colors are mapped to the signal density in the Frequency versus Time bitmap display.
TRACe<x>:FVTime[:MEASview<y>]:COLor:SCALe: AUTO (on page 2583)	Automatically adjusts the Max and Min color settings to display the broadest range of colors in the Frequency versus Time bitmap display.
TRACe:FVTime[:MEASview<y>]:COUNt (on page 2584)	Sets or queries the count for the Max or Min Hold trace.
TRACe:FVTime[:MEASview<y>]:COUNt:ENABle (on page 2584)	Sets or queries whether or not to enable the count for Max or Min Hold.
TRACe:FVTime[:MEASview<y>:COUNt:RESet (on page 2585)	Clears the Max or Min Hold data and counter, and restarts the process.
TRACe<x>:FVTime[:MEASview<y>]:DETection (on page 2585)	Enables or queries the type of detection for the specified trace.
TRACe<x>:FVTime[:MEASview<y>]:DOT: PERSistent (on page 2586)	Determines whether to enable or disable the dot persistence for the bitmap trace (Trace 5) in the Frequency versus Time view.
TRACe<x>:FVTime[:MEASview<y>]:DOT:PERSistent: TYPE (on page 2586)	Sets or queries the persistence type for the bitmap trace (Trace 5) in the Frequency versus Time view.
TRACe<x>:FVTime[:MEASview<y>]:DOT:PERSistent: VARiable (on page 2587)	Sets or queries how long data points are displayed.
TRACe:FVTime[:MEASview<y>]:FREeze (on page 2587)	Sets or queries whether or not to freeze the display of the trace.
TRACe:FVTime[:MEASview<y>]:FUNCtion (on page 2588)	Sets or queries the trace function.
TRACe<x>:FVTime[:MEASview<y>]:SELect (on page 2589)	Sets or queries the selected trace.
TRACe<x>:FVTime[:MEASview<y>]:SELect:LINe (on page 2589)	Sets or queries the line number to send to the Frequency vs time ogram display.
TRACe:IQVTime subgroup	**RF I&Q versus time measurement**
TRACe:IQVTime[:MEASview<y>]:AVERage:COUNt (on page 2590)	Sets or queries the number of traces to combine for averaging.
TRACe:IQVTime[:MEASview<y>]:COUNt (on page 2590)	Sets or queries the count for the Max or Min Hold trace.

Commands	Description
TRACe:IQVTime[:MEASview<y>]:COUNt:ENABle (on page 2591)	Determines whether to enable or disable the count for the Max or Min Hold trace.
TRACe:IQVTime[:MEASview<y>]:DETection (on page 2592)	Sets or queries whether or not to enable the count for Max/Min Hold.
TRACe:IQVTime[:MEASview<y>]:COUNt:RESet (on page 2591)	Clears the Max or Min Hold data and counter, and restarts the process.
TRACe:IQVTime:[MEASview<y>]:ENABle:I (on page 2592)	Sets or queries whether to show or hide the trace I.
TRACe:IQVTime[:MEASview<y>]:ENABle:Q (on page 2593)	Sets or queries whether to show or hide the trace Q.
TRACe:IQVTime[:MEASview<y>]:FREeze (on page 2593)	Sets or queries whether to freeze the IQ traces.
TRACe:IQVTime[:MEASview<y>]:FUNCtion (on page 2594)	Sets or queries the trace function.
TRACe:IQVTime[:MEASview<y>]:SELect:I (on page 2594)	Sets or queries whether to choose the I trace.
TRACe:IQVTime[:MEASview<y>]:SELect:Q (on page 2595)	Sets or queries whether to choose the Q trace.
TRACe:MERRor subgroup	**Magnitude Error measurement**
TRACe:MERRor[:MEASview<y>]:MODE (on page 2596)	Sets or queries whether to display the Magnitude error trace as vectors or symbols.
TRACe:OBWidth subgroup	**Occupied Bandwidth measurement**
TRACe:OBW:MAXHold (on page 2596)	Determines whether to enable or disable the Max Hold trace.
TRACe:OFDM subgroup	**OFDM measurements**
TRACe:OFDM:CONSte (on page 2597)	Determines whether or not to show the trace.
TRACe:OFDM:CONSte:FREeze (on page 2597)	Determines whether or not to freeze the display of the trace.
TRACe<x>:OFDM:EVM (on page 2672)	Determines whether or not to show the specified trace.
TRACe<x>:OFDM:EVM:FREeze (on page 2672)	Determines whether or not to freeze the display of the specified trace.
TRACe<x>:OFDM:EVM:SELect (on page 2673)	Sets or queries the specified trace to display.
TRACe<x>:OFDM:MERRor (on page 2673)	Determines whether or not to show the specified trace.
TRACe<x>:OFDM:MERRor:FREeze (on page 2674)	Determines whether or not to freeze the display of the specified trace.
TRACe<x>:OFDM:MERRor:SELect (on page 2674)	Sets or queries the specified trace to display.
TRACe<x>:OFDM:PERRor (on page 2675)	Determines whether or not to show the specified trace.

Commands	Description
TRACe<x>:OFDM:PERRor:FREeze (on page 2675)	Determines whether or not to freeze the display of the specified trace.
TRACe<x>:OFDM:PERRor:SELect (on page 2676)	Sets or queries the specified trace to display.
TRACe<x>:OFDM:POWer (on page 2676)	Determines whether or not to show the specified trace.
TRACe<x>:OFDM:POWer:FREeze (on page 2677)	Determines whether or not to freeze the display of the specified trace.
TRACe<x>:OFDM:POWer:SELect (on page 2677)	Sets or queries the specified trace to display.
TRACe:OFDM:FLATness:FREeze (on page 2597)	Sets or queries whether to freeze the average trace
TRACe:OFDM:FLATness:SHOW (on page 2598)	Displays or hides the trace on the OFDM Spectral Flatness display.
TRACe:PERRpr subgroup	**Phase Error measurement**
TRACe:PERRor[:MEASview<y>]:MODE (on page 2598)	Sets or queries whether to display the Frequency deviation vs Time trace as vectors or symbols.
TRACe:PHVTime subgroup	**Phase versus time measurement**
TRACe:PHVTime[:MEASview<y>] (on page 2599)	Sets or queries whether or not to show the trace.
TRACe:PHVTime[:MEASview<y>]:AVERage:COUNt (on page 2600)	Sets or queries the number of traces to combine for averaging.
TRACe<x>:PHVTime[:MEASview<y>]:COLor:CURVe (on page 2600)	Sets or queries how colors are mapped to the signal density in the Phase versus Time bitmap display.
TRACe<x>:PHVTime[:MEASview<y>]:COLor:SCALe: AUTO (on page 2601)	Automatically adjusts the Max and Min color settings to display the broadest range of colors in the Phase versus Time bitmap display.
TRACe:PHVTime[:MEASview<y>]:COUNt (on page 2601)	Sets or queries the count for the Max or Min Hold trace.
TRACe:PHVTime[:MEASview<y>]:COUNt:ENABle (on page 2602)	Sets or queries whether or not to enable the count for Max/Min Hold.
TRACe:PHVTime[:MEASview<y>]:COUNt:RESet (on page 2602)	Clears the Max or Min Hold data and counter, and restarts the process.
TRACe<x>:PHVTime[:MEASview<y>]:DETection (on page 2603)	Sets or queries the display detector.
TRACe<x>:PHVTime[:MEASview<y>]:DOT: PERSistent (on page 2603)	Determines whether to enable or disable the dot persistence for the bitmap trace (Trace 5) in the Phase versus Time view.
TRACe<x>:PHVTime[:MEASview<y>]:DOT:PERSistent: TYPE (on page 2604)	Sets or queries the persistence type for the bitmap trace (Trace 5) in the Phase versus Time view.

Commands	Description
TRACe<x>:PHVTime[:MEASview<y>]:DOT:PERSistent: VARiable (on page 2604)	Sets or queries how long data points are displayed.
TRACe<x>:{FSETtling|PSETtling}:FREeze (on page 2653)	Sets or queries whether to freeze the trace.
TRACe:PHVTime[:MEASview<y>]:FREeze (on page 2605)	Determines whether to freeze the trace display in the Phase versus Time measurement.
TRACe:PHVTime[:MEASview<y>]:FUNCtion (on page 2605)	Sets or queries the trace function.
TRACe<x>:PHVTime[:MEASview<y>]:SELect (on page 2606)	Sets or queries the selected trace.
TRACe<x>:PHVTime[:MEASview<y>]:SELect:LINe (on page 2607)	Sets or queries the line number to send to the Phase vs time ogram display.
TRACe:PNOise subgroup (Option 11 only)	**Phase noise measurement**
TRACe<x>:{FSETtling|PSETtling}:SELect (on page 2653)	Selects a trace or queries the currently selected trace.
TRACe<x>:{FSETtling|PSETtling}:SHOW (on page 2654)	Sets or queries whether to show or hide the trace.
TRACe:{FSETtling|PSETtling}:AVERage:ENABle (on page 2580)	Sets or queries the number of data points for smoothing the trace.
TRACe<x>:PNOise:SELect (on page 2682)	Selects the trace in the phase noise measurement.
TRACe<x>:PNOise:SHOW (on page 2683)	Shows or hides the specified trace in the phase noise view.
TRACe<x>:PNOise:SMOothing:COUNt (on page 2683)	Sets or queries the number of data points to take the moving average for smoothing the trace.
TRACe<x>:PNOise:SMOothing:ENABle (on page 2683)	Sets or queries whether to enable smoothing the specified trace.
TRACe<x>:PNOise:SMOothing:RESet (on page 2684)	Restarts the smoothing process.
TRACe:{FSETtling|PSETtling} subgroup (Option 12 only)	**Settling time measurements**
TRACe:{FSETtling|PSETtling}:AVERage:COUNt (on page 2579)	Sets or queries the number of traces to combine.
TRACe:{FSETtling|PSETtling}:AVERage:COUNt: CURRent? (on page 2580)	Queries the current running average count value.
TRACe:{FSETtling|PSETtling}:AVERage:ENABle (on page 2580)	Determines whether to enable or disable averaging the trace(s).
TRACe:{FSETtling|PSETtling}:RESet (on page 2580)	If trace averaging is enabled, this command resets the current averaged trace to 0.
TRACe:{FSETtling|PSETtling}:SMOothing:COUNt (on page 2581)	Sets or queries the number of data points to take the moving average for smoothing the traces.
TRACe:{FSETtling|PSETtling}:SMOothing:ENABle (on page 2581)	Determines whether to enable or disable smoothing the trace(s).

Commands	Description
TRACe<x>:{FSETtling|PSETtling}:FREeze (on page 2653)	Determines whether to freeze the specified trace in the Settling Time measurement display.
TRACe<x>:{FSETtling|PSETtling}:SELect (on page 2653)	Selects the trace in the settling time measurement.
TRACe<x>:{FSETtling|PSETtling}:SHOW (on page 2654)	Determines whether to show or hide the specified trace in the settling time view.
TRACe:SEM subgroup	**Spectral Emissions Mask measurements**
TRACe:SEM:COUNt (on page 2607)	Sets or queries how many acquisitions run in the single acquisition mode for multi-trace functions.
TRACe:SEM:COUNt:RESet (on page 2608)	Clears the multi-function data and counter and restarts the process.
TRACe:SEM:FUNCtion (on page 2608)	Sets or queries the trace function.
TRACe:SGRam subgroup	**Spectrogram measurement**
TRACe:SGRam[:MEAview<y>]:DETection (on page 2609)	Sets or queries the method to be used for decimating traces to fit the on screen.
TRACe:SGRam[:MEAview<y>]:FREeze (on page 2609)	Sets or queries whether or not to freeze the spectrogram display.
TRACe:SGRam[:MEAview<y>]:FUNCtion (on page 2610)	Sets or queries the trace function for the specified trace in the spectrogram.
TRACe:SGRam[:MEAview<y>]:FUNCtion:TIME (on page 2611)	Sets or queries the number of traces to combine for the trace function.
TRACe:SGRam[:MEAview<y>]:SELect:LINE (on page 2612)	Sets or queries the line number to send to the spectrum display.
TRACe<x>:SPECtrum subgroup	**Spectrum measurement**
TRACe<x>:SPECtrum[:MEAview<y>] (on page 2684)	Sets or queries whether to show or hide the specified trace.
TRACe<x>:SPECtrum[:MEAview<y>]:AVERage: COUNt (on page 2685)	Sets or queries the number of traces to combine for averaging.
TRACe<x>:SPECtrum[:MEAview<y>]:AVERage: RESet (on page 2685)	Clears the average data and resets the average counter.
TRACe<x>:SPECtrum[:MEASview<y>]:COLor:CURVe (on page 2686)	Sets or queries how colors are mapped to the signal density in the Spectrum bitmap display.
TRACe<x>:SPECtrum[:MEASview<y>]:COLor:SCALe: AUTO (on page 2687)	Automatically adjusts the Max and Min color settings to display the broadest range of colors in the Spectrum bitmap display.
TRACe<x>:SPECtrum[:MEAview<y>]:COUNt (on page 2687)	Sets or queries the count for the Max or Min Hold trace.
TRACe<x>:SPECtrum[:MEAview<y>]:COUNt: ENABle (on page 2688)	Sets or queries whether or not to enable the count for Max/Min Hold.

Commands	Description
TRACe<x>:SPECtrum[:MEAview<y>]:COUNt:RESet (on page 2688)	Clears the Max or Min Hold data and counter, and restarts the process.
TRACe<x>:SPECtrum[:MEAview<y>]:DETection (on page 2689)	Sets or queries the display detector.
TRACe<x>:SPECtrum[:MEASview<y>]:DOT:PERSistent (on page 2689)	Determines whether to enable or disable the dot persistence for the bitmap trace (Trace 6) in the Spectrum view.
TRACe<x>:SPECtrum[:MEASview<y>]:DOT:PERSistent: TYPE (on page 2690)	Sets or queries the persistence type for the bitmap trace (Trace 6) in the Spectrum view.
TRACe<x>:SPECtrum[:MEASview<y>]:DOT:PERSistent: VARiable (on page 2690)	Sets or queries how long data points are displayed.
TRACe<x>:SPECtrum[:MEAview<y>]:FREeze (on page 2691)	Sets or queries whether or not to freeze the display of the trace.
TRACe<x>:SPECtrum[:MEAview<y>]:FUNCtion (on page 2691)	Sets or queries the trace function.
TRACe<x>:SPECtrum[:MEAview<y>]:LEFToperand (on page 2692)	Sets or queries the left operand for the math trace.
TRACe<x>:SPECtrum[:MEAview<y>]:RIGHtoperand (on page 2693)	Sets or queries the right operand for the math trace.
TRACe<x>:SPECtrum[:MEAview<y>]:SELect (on page 2693)	Sets or queries the trace number to display the readout.
TRACe:SPURious subgroup	**Spurious measurement**
TRACe:SPURious:COUNt (on page 2612)	Sets or queries the count for the Max Hold or Average trace.
TRACe:SPURious:COUNt:ENABle (on page 2613)	Sets or queries whether to enable the count for Max Hold or Average.
TRACe:SPURious:COUNt:RESet (on page 2613)	Clears Max Hold or Average data and counter, and restarts the process.
TRACe:SPURious:FREeze (on page 2613)	Determines whether or not to freeze the display of the trace.
TRACe:SPURious:FUNCtion (on page 2614)	Sets or queries the trace function.
TRACe:TOVerview subgroup	**All measurements**
TRACe1:TOVerview[:MEASview<y>] (on page 2616)	Enables display of or queries the display status of the specified trace.
TRACe1:TOVerview[:MEASview<y>]:AVERage: COUNt (on page 2617)	Sets or queries the number of traces averaged to generate the specified trace.
TRACe1:TOVerview[:MEASview<y>]:COUNt (on page 2617)	Enables or queries the count set for the specified trace.
TRACe1:TOVerview[:MEASview<y>]:COUNt:ENABle (on page 2618)	Enables or queries the Average count for the specified trace.

Commands	Description
TRACe1:TOVerview[:MEASview<y>]:COUNt:RESet (on page 2618)	Sets or queries whether or not to freeze the display of the trace.
TRACe1:TOVerview[:MEASview<y>]:DETection (on page 2619)	Enables or queries the type of detection for the specified trace.
TRACe1:TOVerview[:MEASview<y>]:FREeze (on page 2619)	Enables or queries a halt to acquisition updates for the specified trace.
TRACe1:TOVerview[:MEASview<y>]:FUNCTion (on page 2620)	Enables or queries the selected Function for the specified trace.
TRACe:WLAN group	**Wireless LAN measurements**
TRACe:WLAN:FLATness:FREeze (on page 2615)	Sets or queries whether to freeze the average trace on the WLAN Spectral Flatness display.
TRACe:WLAN:FLATness:SHOW (on page 2616)	Displays or hides the trace on the WLAN Spectral Flatness display.
TRACe<x>:WLAN:CONSte (on page 2700)	Turns the selected trace on or off in the Constellation plot display.
TRACe<x>:WLAN:CONSte:FREeze (on page 2700)	Turns the freeze state of the selected trace on or off.
TRACe<x>:WLAN:CRESponse:SELect (on page 2701)	Sets or queries the active trace in the WLAN Channel Response.
TRACe<x>:WLAN:EVM:FREeze (on page 2701)	Turns the freeze state of the selected trace on or off.
TRACe<x>:WLAN:EVM[:SHOW] (on page 2702)	Displays or hides the selected trace.
TRACe<x>:WLAN:MERRor:FREeze (on page 2702)	Turns the freeze state of the selected trace on or off.
TRACe<x>:WLAN:MERRor[:SHOW] (on page 2703)	Displays or hides the selected trace.
TRACe<x>:WLAN:PERRor:FREeze (on page 2703)	Turns the freeze state of the selected trace on or off.
TRACe<x>:WLAN:PERRor[:SHOW] (on page 2704)	Displays or hides the selected trace.
TRACe:LTE subgroup	
TRACe<x>:LTE:ACLR:COUNt (on page 2654)	Sets or queries how many acquisitions run in the single acquisition mode for multitrace functions.
TRACe<x>:LTE:ACLR:COUNt:RESet (on page 2655)	Clears the multifunction (Max Hold or Average) data and counter and restarts the process.
TRACe<x>:LTE:ACLR:FUNCtion (on page 2655)	Sets or queries the trace function.
TRACe<x>:LTE:CONSte:SYNChronization: SEQUence:STATe? (on page 2656)	Returns whether or not a measurement signal (Primary Synchronization Signal/Secondary Synchronization Signal) is selected for the specified trace.

Trace mnemonics

Trace Mnemonics

Multiple traces can be used in some measurement displays. The traces are specified by the trace specifier TRACe<x> (<x>=1 to 5) which is defined for each measurement display as follows.

Trace mnemonics

Measurement display	TRACe1	TRACe2	TRACe3	TRACe4	TRACe5
Amplitude versus Time	Trace 1	Trace 2	Math trace	NA	NA
Phase noise	Trace 1	Trace 2	NA	NA	NA
Spectrum	Trace 1	Trace 2	Trace 3	Math trace	Spectrogram

NOTE: Valid traces depend on commands. Refer to each command description.

Trigger commands

Use the TRIGger commands to set up the trigger system.

Trigger commands

Commands	Description
*TRG (No Query Form) (on page 2704)	Generates a trigger. It produces the same effect as the Force Trigger button on the Trigger control panel. This command is valid when the trigger mode is Triggered.
TRIGger:ADVanced:SWEep:MODE (on page 2705)	Sets or enables the analyzer to wait for a trigger for each acquisition in a swept spectrum or other measurement requiring multiple acquisitions worth of span.
TRIGger:DPX:SHOW:FRAMes (on page 2705)	Sets or queries whether to trigger each segment in the swept mode.
TRIGger:DPX:TDM:SHOW:ALL (on page 2706)	Sets or queries the selections of the DPX Prefs control panel tab page.
TRIGger:EVENt:EXTernal:SLOPe (on page 2706)	Sets or queries the signal slope of the External input to determine a trigger event.
TRIGger:EVENt:INPut:LEVel (on page 2707)	Sets or queries the power level that the signal must pass through to cause a trigger event when the trigger type is set to Power.
TRIGger:EVENt:INPut:SLOPe (on page 2707)	Sets or queries the signal slope of the RF Input to determine a trigger event.
TRIGger:EVENt:INPut:TYPE (on page 2707)	Sets or queries the type of trigger. (Power is the only available setting.)
TRIGger:EVENt:SOURce (on page 2708)	Sets or queries the source of the signal to be monitored for a trigger event.
TRIGger:MASK:NEW (on page 2708)	Loads a new frequency mask.
TRIGger:MASK:NEW:AUTO (on page 2709)	Draws a new frequency mask automatically.

Commands	Description
TRIGger:MASK:OPEN (on page 2710)	Opens a trigger mask with a specified file.
TRIGger:MASK:SAVE (on page 2711)	Saves the current trigger mask to a specified file.
TRIGger:SAVE:COUNt:MAXimum (on page 2711)	Sets or queries the number of times a Save on Trigger action stores a file.
TRIGger:SAVE:COUNt[:STATe] (on page 2711)	Sets or queries the Maximum total number of files feature.
TRIGger:SAVE:DATA:FORMat (on page 2712)	Sets or queries the file format used when saving the acquisition data on a trigger event.
TRIGger:SAVE:DATA[:STATe] (on page 2712)	Sets or queries to save data when a trigger event occurs.
TRIGger:SAVE:PICTure:FORMat (on page 2713)	Sets or queries the file format used when saving a picture of the display when a trigger event occurs.
TRIGger:SAVE:PICTure[:STATe] (on page 2713)	Sets or queries the save picture on trigger setting.
TRIGger[:SEQuence]:ADVanced:HOLDoff (on page 2713)	FIRST Sets or queries the trigger holdoff time.
TRIGger[:SEQuence]:ADVanced:HOLDoff:ENABle (on page 2714)	Sets or queries the trigger delay time.
TRIGger[:SEQuence]:ADVanced:SWEep:MODE (on page 2714)	Determines whether or not to trigger each segment in the swept acquisition mode.
TRIGger[:SEQuence]:ADVanced:SWEPt:SEGMent:ENABle (on page 2715)	Determines whether or not to wait for a trigger for each acquisition in a swept spectrum.
TRIGger[:SEQuence]:EVENt:A:EDGE:COUPling (on page 2715)	Sets or queries the choice of Trigger coupling type.
TRIGger[:SEQuence]:EVENt:A:EDGE:HOLDoff:MODE (on page 2716)	Sets or queries the trigger hold off mode.
TRIGger[:SEQuence]:EVENt:A:EDGE:HOLDoff:TIME (on page 2716)	Sets or queries the hold off time.
TRIGger[:SEQuence]:EVENt:A:EDGE:LEVEl (on page 2716)	Sets or queries where the trigger point occurs for trigger source type.
TRIGger[:SEQuence]:EVENt:EXTFront:IMPedance (on page 2717)	Sets or queries the impedance of the external trigger input (front).
TRIGger[:SEQuence]:EVENt:EXTFront:LEVel (on page 2717)	Sets or queries the trigger level at the external trigger input (front).
TRIGger[:SEQuence]:EVENt:EXTFront:SLOPe (on page 2718)	Sets or queries the trigger slope of the external trigger input (front).

Commands	Description
TRIGger[:SEQuence]:EVENt:EXTRear:SLOPe (on page 2718)	Sets or queries the trigger slope of the external trigger input (rear).
TRIGger[:SEQuence]:EVENt:GATed (on page 2718)	Sets or queries the logic for the gated trigger input.
TRIGger[:SEQuence]:EVENt:INPut:DDENsity:AMPLitude (on page 2719)	Sets or queries the center vertical level for the density trigger region.
TRIGger[:SEQuence]:EVENt:INPut:DDENsity:AMPLitude:TOLerance (on page 2719)	Sets or queries the trigger level for the RF input level trigger.
TRIGger[:SEQuence]:EVENt:INPut:DDENsity:FREQuency (on page 2720)	Sets or queries the center horizontal level for the density trigger region.
TRIGger[:SEQuence]:EVENt:INPut:DDENsity:FREQuency:TOLerance (on page 2720)	Sets or queries the left and right extensions of the density trigger region from the center horizontal level.
TRIGger[:SEQuence]:EVENt:INPut:DDENsity:THReshold (on page 2721)	Sets or queries the DPX Density threshold that defines a trigger event.
TRIGger[:SEQuence]:EVENt:INPut:DDENsity:VIOLation (on page 2721)	Sets or queries whether a density value higher or lower than the THReshold value defines a trigger event.
TRIGger[:SEQuence]:EVENt:INPut:FEDGe:LEVel (on page 2722)	Sets or queries the trigger level at the frequency edge input.
TRIGger[:SEQuence]:EVENt:INPut:FEDGe:LEVel:ACTual? (on page 2722)	Queries the actual length of the frequency edge input for the DPX measurement.
TRIGger[:SEQuence]:EVENt:INPut:FEDGe:SLOPe (on page 2723)	Sets or queries the trigger slope of the frequency edge input.
TRIGger[:SEQuence]:EVENt:INPut:FEDGe:THReshold:LEVel (on page 2723)	Set or queries the threshold level in the DPX measurement.
TRIGger[:SEQuence]:EVENt:INPut:FEDGe:THReshold:STATe (on page 2724)	Set or queries the threshold state in the DPX measurement.
TRIGger[:SEQuence]:EVENt:INPut:FMASk:{ BANDwidth|BWIDth}[:RESolution] (on page 2724)	Sets or queries the Resolution Bandwidth value to be used in thespectrum view for the frequency mask trigger.
TRIGger[:SEQuence]:EVENt:INPut:FMASk:BANDwidth|BWIDth[:RESolution]:ACTual? (on page 2725)	Sets or queries when the trigger occurs in the frequency mask trigger.
TRIGger[:SEQuence]:EVENt:INPut:FMASk:BANDwidth|BWIDth[:RESolution]:AUTO (on page 2725)	Sets or queries whether to automatically set the Resolution Bandwidth value used in the spectrum view for the frequency mask trigger.
TRIGger[:SEQuence]:EVENt:INPut:FMASk:VIOLation (on page 2725)	Sets or queries when the analyzer triggers in the frequency mask trigger.

Commands	Description
TRIGger[:SEQuence]:EVENt:INPut:LEVel (on page 2726)	Sets or queries the trigger level for the RF input level trigger.
TRIGger[:SEQuence]:EVENt:INPut:RUNT:PULSe (on page 2727)	Sets or queries the Runt trigger for a positive or negative going pulse.
TRIGger[:SEQuence]:EVENt:INPut:RUNT:PULse:HIGH:LEVel (on page 2727)	Sets or queries the trigger slope for the RF input level trigger.
TRIGger[:SEQuence]:EVENt:INPut:RUNT:PULSe:LOW:LEVel (on page 2728)	Sets or queries the lower level (second threshold) to qualify a Runt trigger.
TRIGger[:SEQuence]:EVENt:INPut:SLOPe (on page 2728)	Sets or queries the trigger slope for the RF input level trigger.
TRIGger[:SEQuence]:EVENt:INPut:TDBWidth (on page 2728)	Sets or queries the time-domain bandwidth.
TRIGger[:SEQuence]:EVENt:INPut:TDBWidth:ACTual? (on page 2729)	Queries the actual time-domain bandwidth.
TRIGger[:SEQuence]:EVENt:INPut:TDBWidth:STATe (on page 2729)	Determines whether to set the time-domain bandwidth automatically.
TRIGger[:SEQuence]:EVENt:INPut:TYPE (on page 2730)	Sets or queries the trigger type for the source of RF input.
TRIGger[:SEQuence]:EVENt:SOURce (on page 2730)	Sets or queries the trigger event source.
TRIGger[:SEQuence]:FORCed (on page 2731)	Sets or queries whether to causes a manual trigger.
TRIGger:SEQuence:FORCed (No Query Form) (on page 2731)	Causes a trigger event to occur when the trigger mode is set to Triggered, regardless of the set trigger conditions.
TRIGger[:SEQuence]:IMMediate (on page 2732)	Causes a trigger immediately.
TRIGger[:SEQuence]:STATus (on page 2732)	Sets or queries the trigger mode (Free Run or Triggered).
TRIGger[:SEQuence][:STATus] (on page 2733)	Sets or queries the trigger mode setting.
TRIGger[:SEQuence]:TIME:DELay (on page 2733)	Sets or queries the trigger delay time.
TRIGger[:SEQuence]:TIME:POSition (on page 2733)	Sets or queries the trigger position.
TRIGger[:Sequence]:TIME:POSition:AUTO (on page 2734)	Sets or queries the position of the trigger event (automatic or manual) within the acquisition record.
TRIGger[:SEQuence]:TIME:QUALified (on page 2734)	LAST Sets or queries the timing qualification setting for triggers.

Commands	Description
TRIGger[:SEQuence]:TIME:QUALified:TIME<x> (on page 2735)	Sets or queries the trigger mode (Free Run or Triggered).
TRIGger[:SEQuence]:EVENt:A:EDGE:SETTings (on page 2735)	Sets or queries the settings of Trigger source type.
TRIGger[:SEQuence]:EVENt:A:EDGE:SLOPe (on page 2736)	Sets or queries whether instrument finds the trigger point on rising or falling edge or either edge of a signal.
TRIGger[:SEQuence]:EVENt:A:EDGE:SOURce (on page 2736)	Sets or queries the channel on which signal is to be monitored for a trigger event.
TRIGger[:SEQuence:]EVENt:A:TYPE (on page 2737)	Sets or queries the trigger type {supporting only Edge trigger}.
TRIGger[:SEQuence]:MODE (on page 2737)	Sets or queries the trigger mode setting.
TRIGger:TIME:POSition (on page 2737)	Sets or queries the point (in percent) along the acquisition record the trigger event should occur.
TRIGger:TIME:POSition:AUTO (on page 2738)	Sets or queries the instrument to automatically set the position of the trigger event.

Unit commands

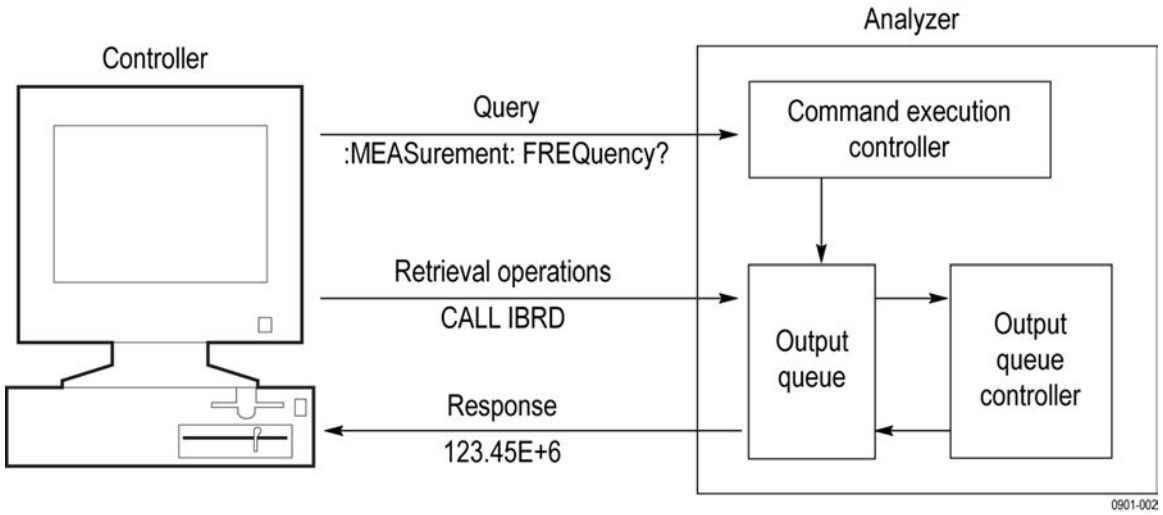
Specify fundamental units for measurement.

Unit commands

Header	Description
UNIT:POWer (on page 2738)	Sets or queries the unit of power.

Retrieving response message

When receiving a query command from the external controller, the analyzer puts the response message on the Output Queue. This message cannot be retrieved unless you perform retrieval operations through the external controller. (For example, call the IBRD subroutine included in the GPIB software of National Instruments.)



Retrieving response message

When the Output Queue contains a response message, sending another command from the external controller before retrieving this message deletes it from the queue. The Output Queue always contains the response message to the most recent query command.

You can use the MAV bit of the Status Byte Register (SBR) to check whether the Output Queue contains a response message. For details, refer to Status Byte (on page 2742).

Command descriptions

ABORt (No query form)

Resets the trigger system and places all trigger sequences in the idle state. Any actions related to the trigger system that are in progress, such as a sweep or acquiring a measurement is also aborted.

To start data acquisition, use the INITiate commands.

Conditions

Measurement views: All

Group

Abort commands

Syntax

`ABORt`

Related Commands

INITiate:CONTinuous (on page 1480)

INITiate:RESume (on page 1481)

Arguments

None

Examples

`ABORt` resets the trigger system and stops data acquisition.

*CAL (Query only)

Conditions

Measurement views: All

Group

IEEE common commands

Syntax

`*CAL`

Returns

`<NR1>=1` indicates that the alignment was successful.

`<NR1>=0` indicates that the alignment was unsuccessful.

Examples

*CAL performs an internal self-alignment and will return 1 if the alignment is successful.

Calculate commands

CALCulate:[:SEARch|:TEST]:LIMit[:STATe]:MATCh:FILE:LOCation

Sets or queries the destination folder for “search match” output files.

Conditions

Measurement views: All

Group

Calculate commands

Syntax

```
CALCulate:[:SEARch|:TEST]:LIMit[:STATe]:MATCh:FILE:LOCation <string>
CALCulate:[:SEARch|:TEST]:LIMit[:STATe]:MATCh:FILE:LOCation?
```

Arguments

<string> specifies the destination folder.

Returns

<string>

Examples

CALCULATE:[:SEARCH|:TEST]:LIMIT[:STATE]:MATCH:FILE:LOCATION? might return c:\rsamap files, indicating the destination folder.

CALCulate[:SEARch|:TEST]:LIMit[:STATe]:MATCh:FILE:NAME:BASE

Sets or queries the file basename for “search match” output files.

Conditions

Measurement views: All

Group

Calculate commands

Syntax

```
CALCulate:[:SEARch|:TEST]:LIMit[:STATe]:MATCh:FILE:NAME:BASE<string>
CALCulate:[:SEARch|:TEST]:LIMit[:STATe]:MATCh:FILE:NAME:BASE?
```

Arguments

<string> specifies the output file.

Returns

<string>

Examples

CALCULATE:SEARCH:LIMIT:STATE:MATCH:FILE:NAME:BASE "BASE" sets the base filename as BASE

CALCulate:ACPower:MARKer<x>:DELTa:X? (Query Only)

Returns the delta marker frequency for the selected marker in the Channel power and ACPR measurement.

The parameter <x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: Channel power and ACPR

Group

Calculate commands

Syntax

CALCulate:ACPower:MARKer<x>:DELTa:X?

Related Commands

CALCulate:MARKer:ADD (on page 458)

CALCulate:ACPower:MARKer<x>:DELTa:Y? (on page 344)

Returns

<NRf> Delta marker frequency for the selected marker.

Examples

CALCULATE:ACPOWER:MARKER1:DELTA:X? might return 1.28E+6, indicating that the delta marker frequency is 1.28 MHz.

CALCulate:ACPower:MARKer<x>:DELTa:Y? (Query Only)

Returns the delta marker amplitude for the selected marker in the Channel power and ACPR measurement.

The parameter <x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: Channel power and ACPR

Group

Calculate commands

Syntax

CALCulate:ACPower:MARKer<x>:DELTa:Y?

Related Commands

CALCulate:MARKer:ADD (on page 458)

CALCulate:ACPower:MARKer<x>:DELTa:X? (on page 344)

Returns

<NRf> Delta marker amplitude for the selected marker.

Examples

CALCULATE:ACPOWER:MARKER1:DELTA:Y? might return 23.45, indicating that the delta marker amplitude is 23.45 dB.

CALCulate:ACPower:MARKer<x>:MAXimum (No Query Form)

Moves the selected marker to the highest peak on the trace in the Channel power and ACPR measurement

Conditions

Measurement views: Channel power and ACPR

Group

Calculate commands

Syntax

CALCulate:ACPower:MARKer<x>:MAXimum

Arguments

None

Examples

CALCULATE:ACPOWER:MARKER1:MAXIMUM moves Marker 1 (M1) to the highest peak on the trace.

CALCulate:ACPower:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the selected marker to the next peak to the left on the trace in the Channel power and ACPR measurement.

Conditions

Measurement views: Channel power and ACPR

Group

Calculate commands

Syntax

CALCulate:ACPower:MARKer<x>:PEAK:LEFT

Related Commands

CALCulate:ACPower:MARKer<x>:PEAK:RIGHt (on page 345)

Arguments

None

Examples

CALCULATE:ACPOWER:MARKER1:PEAK:LEFT moves Marker 1 (M1) to the next peak to the left on the trace.

CALCulate:ACPower:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the selected marker to the next peak to the right on the trace in the Channel power and ACPR measurement.

Conditions

Measurement views: Channel power and ACPR

Group

Calculate commands

Syntax

```
CALCulate:ACPower:MARKer<x>:PEAK:RIGHt
```

Related Commands

CALCulate:ACPower:MARKer<x>:PEAK:LEFT (on page 345)

Arguments

None

Examples

`CALCULATE:ACPOWER:MARKER1:PEAK:RIGHT` moves Marker 1 (M1) to the next peak to the right on the trace.

CALCulate:ACPower:MARKer<x>:X

Sets or queries the horizontal position of the selected marker in the Channel power and ACPR measurement.

Conditions

Measurement views: Channel power and ACPR

Group

Calculate commands

Syntax

```
CALCulate:ACPower:MARKer<x>:X <value>
CALCulate:ACPower:MARKer<x>:X?
```

Related Commands

CALCulate:ACPower:MARKer<x>:Y? (on page 346)

Arguments

`<value>::=<NRf>` specifies the horizontal position of the marker. Range: Start to Stop frequency (left to right edge of the horizontal axis). Using an out-of-range value causes an execution error (-222, "Data out of range").

Examples

`CALCULATE:ACPOWER:MARKER1:X 800MHz` places Marker 1 (M1) at 800 MHz on the trace.

CALCulate:ACPower:MARKer<x>:Y? (Query Only)

Queries the marker amplitude of the selected marker in the Channel power and ACPR measurement.

Conditions

Measurement views: Channel power and ACPR

Group

Calculate commands

Syntax

CALCulate:ACPower:MARKer<x>:Y?

Related Commands

CALCulate:ACPower:MARKer<x>:X (on page 346)

Arguments

None

Returns

<NRf> Marker amplitude of the selected marker.

Examples

CALCULATE:ACPOWER:MARKER1:Y might return -34.28, indicating Marker 1 (M1) is at -34.28 dBm.

CALCulate:{AM|FM|PM}:MARKer<x>:DELTa:X? (Query Only)

Returns the delta marker time for the selected marker in the AM/FM/PM measurement.

The parameter <x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: General purpose analog demodulation

Group

Calculate commands

Syntax

CALCulate:{AM|FM|PM}:MARKer<x>:DELTa:X?

Related Commands

CALCulate:{AM|FM|PM}:MARKer<x>:DELTa:Y? (on page 347)

Arguments

None

Returns

<NRf> Delta marker time for the selected marker.

Examples

CALCULATE:AM:MARKER1:DELTA:X? might return 38.0E-6, indicating that the delta marker time is 38.0 µs.

CALCulate:{AM|FM|PM}:MARKer<x>:DELTa:Y? (Query Only)

Returns the delta marker amplitude for the selected marker in the AM/FM/PM measurement.

The parameter <x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: General purpose analog demodulation

Group

Calculate commands

Syntax

```
CALCulate:{AM|FM|PM}:MARKer<x>:DELTa:Y?
```

Related Commands

CALCulate:{AM|FM|PM}:MARKer<x>:DELTa:X? (on page 347)

Arguments

None

Returns

<NRf> Delta marker amplitude for the selected marker, indicating the difference of modulation factor in percent (AM), frequency deviation in Hz (FM), or phase deviation in degrees (PM) with the reference marker.

Examples

CALCULATE:AM:MARKER1:DELTA:Y? might return 45.82, indicating that the delta marker amplitude is the modulation factor difference of 45.82% in the AM measurement.

CALCulate:{AM|FM|PM}:MARKer<x>:MAXimum (No Query Form)

Moves the selected marker to the highest peak on the trace in the AM/FM/PM measurement.

Conditions

Measurement views: General purpose analog demodulation

Group

Calculate commands

Syntax

```
CALCulate:{AM|FM|PM}:MARKer<x>:MAXimum
```

Arguments

None

Examples

CALCULATE:AM:MARKER1:MAXIMUM moves Marker 1 (M1) to the highest peak on the AM-demodulated trace.

CALCulate:{AM|FM|PM}:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the selected marker to the next peak higher in amplitude on the trace in the AM/FM/PM measurement.

Conditions

Measurement views: General purpose analog demodulation

Group

Calculate commands

Syntax

```
CALCulate:{AM|FM|PM}:MARKer<x>:PEAK:HIGHer
```

Related Commands

CALCulate:{AM|FM|PM}:MARKer<x>:PEAK:LOWer (on page 349)

Arguments

None

Examples

CALCULATE:AM:MARKER1:PEAK:HIGHER moves Marker 1 (M1) to the next peak higher in amplitude on the AM-demodulated trace.

CALCulate:{AM|FM|PM}:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the selected marker to the next peak to the left on the trace in the AM/FM/PM measurement.

Conditions

Measurement views: General purpose analog demodulation

Group

Calculate commands

Syntax

CALCulate:{AM|FM|PM}:MARKer<x>:PEAK:LEFT

Related Commands

CALCulate:{AM|FM|PM}:MARKer<x>:PEAK:RIGHt (on page 350)

Arguments

None

Examples

CALCULATE:AM:MARKER1:PEAK:LEFT moves Marker 1 (M1) to the next peak to the left on the AM-demodulated trace.

CALCulate:{AM|FM|PM}:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the selected marker to the next peak lower in amplitude on the trace in the AM/FM/PM measurement.

Conditions

Measurement views: General purpose analog demodulation

Group

Calculate commands

Syntax

CALCulate:{AM|FM|PM}:MARKer<x>:PEAK:LOWer

Related Commands

CALCulate:{AM|FM|PM}:MARKer<x>:PEAK:HIGHer (on page 348)

Arguments

None

Examples

CALCULATE:AM:MARKER1:PEAK:LOWER moves Marker 1 (M1) to the next peak lower in amplitude on the AM-demodulated trace.

CALCulate:{AM|FM|PM}:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the selected marker to the next peak to the right on the trace in the AM/FM/PM measurement.

Conditions

Measurement views: General purpose analog demodulation

Group

Calculate commands

Syntax

CALCulate:{AM|FM|PM}:MARKer<x>:PEAK:RIGHt

Related Commands

CALCulate:{AM|FM|PM}:MARKer<x>:PEAK:LEFT (on page 349)

Arguments

None

Examples

CALCULATE:AM:MARKER1:PEAK:RIGHT moves Marker 1 (M1) to the next peak to the right on the AM-demodulated trace.

CALCulate:{AM|FM|PM}:MARKer<x>:X

Sets or queries the horizontal position (time) of the selected marker in the AM/FM/PM measurement.

Conditions

Measurement views: General purpose analog demodulation

Group

Calculate commands

Syntax

CALCulate:{AM|FM|PM}:MARKer<x>:X <value>
CALCulate:{AM|FM|PM}:MARKer<x>:X?

Related Commands

CALCulate:{AM|FM|PM}:MARKer<x>:Y? (on page 351)

Arguments

<value>::=<NRf> specifies the horizontal position of the marker. Range: (analysis offset) to [(analysis offset) + (analysis length)].

Examples

CALCULATE:AM:MARKER1:X 1.5us places Marker 1 (M1) at 1.5 µs on the AM-demodulated trace.

CALCulate:{AM|FM|PM}:MARKer<x>:Y? (Query Only)

Queries the marker amplitude of the selected marker in the AM/FM/PM measurement.

Conditions

Measurement views: General purpose analog demodulation

Group

Calculate commands

Syntax

CALCulate:{AM|FM|PM}:MARKer<x>:Y?

Related Commands

CALCulate:{AM|FM|PM}:MARKer<x>:X (on page 350)

Arguments

None

Returns

<NRf> Marker amplitude of the selected marker, indicating the modulation factor in percent (AM), frequency deviation in Hz (FM), or phase deviation in degrees (PM) at the marker.

Examples

CALCULATE:AM:MARKER1:Y? might return 23.4, indicating that Marker 1 (M1) reads the modulation factor of 23.4% in the AM measurement.

CALCulate:AUDio:HARMonic:HNUMber

Sets or queries the number of harmonics used for the audio measurement. The range of values is an integer from 1 to 20.

Conditions

Measurement views: Audio Spectrum

Group

Calculate commands

Syntax

CALCulate:AUDio:HARMonic:HNUMber <value>
CALCulate:AUDio:HARMonic:HNUMber?

Arguments

<value> ::= <NR1> specifies the number of harmonics.

Examples

CALCULATE:AUDIO:HARMONIC:HNUMBER 20 sets the number of harmonics to 20.

CALCulate:AUDio:HARMonic:NHNumber

Sets or queries the number of non-harmonics used for the audio measurement. The range of values is an integer from 0 to 20.

Conditions

Measurement views: Audio Spectrum

Group

Calculate commands

Syntax

```
CALCulate:AUDio:HARMonic:NHNumber <value>
CALCulate:AUDio:HARMonic:NHNumber?
```

Arguments

`<value> ::= <NR1>` specifies the number of non-harmonics.

Examples

`CALCULATE:AUDIO:HARMONIC:NHNUMBER 20` sets the number of non-harmonics to 20.

CALCulate:AUDio:NHARmonic:EXCursion

Sets or queries the non-harmonic excursion value. The range of values is 0 dB to 30 dB.

Conditions

Measurement views: Audio Spectrum

Group

Calculate commands

Syntax

```
CALCulate:AUDio:NHARmonic:EXCursion <value>
CALCulate:AUDio:NHARmonic:EXCursion?
```

Arguments

`<value> ::= <NR1>` The specifies the non-harmonic excursion value.

Examples

`CALCULATE:AUDIO:NHARMONIC:EXCURSION 6` sets the non-harmonic excursion to 6.0 dB.

CALCulate:AUDio:NHARmonic:IGNore

Sets or queries the ignore region of the non-harmonics in the audio measurement. The range of the ignore region is from 0 Hz to the audio bandwidth.

Conditions

Measurement views: Audio Spectrum

Group

Calculate commands

Syntax

```
CALCulate:AUDio:NHARmonic:IGNore <value>
CALCulate:AUDio:NHARmonic:IGNore?
```

Arguments

`<value> ::= <NRf>` specifies the ignore region.

Examples

`CALCULATE:AUDIO:NHARMONIC:IGNORE 1E+3` sets the non-harmonic ignore region to 1.0 kHz.

CALCulate:AUDio:NHARmonic:THReshold

Sets or queries the non-harmonic threshold value for the audio measurement. The range is from -200 dBc to 0 dBc.

Conditions

Measurement views: Audio Spectrum

Group

Calculate commands

Syntax

```
CALCulate:AUDio:NHARmonic:THReshold <value>
CALCulate:AUDio:NHARmonic:THReshold?
```

Arguments

`<value> ::= <NR1>` specifies the threshold value.

Examples

`CALCULATE:AUDIO:NHARMONIC:THRESHOLD -100` sets the non-harmonic threshold value to -100 dBc.

CALCulate:AUDio:SPECtrum:MARKer<x>:DELTa:X? (Query Only)

Returns the delta marker frequency for the selected marker on the spectrum trace.

The parameter <x> = 1 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Conditions

Measurement views: Audio Spectrum

Group

Calculate commands

Syntax

```
CALCulate:AUDio:SPECtrum:MARKer<x>:DELTa:X?
```

Arguments

None

Returns

`<NRf>` the delta marker frequency for the selected marker.

Examples

`CALCulate:AUDio:SPECtrum:MARKer1:DELTa:X?` might return 5.0020E+3 indicating the delta marker frequency is 5.002 kHz.

CALCulate:AUDio:SPECtrum:MARKer<x>:DELTa:Y? (Query Only)

Returns the delta marker amplitude for the selected marker on the spectrum trace.

The parameter <x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: Audio Spectrum

Group

Calculate commands

Syntax

`CALCulate:AUDio:SPECtrum:MARKer<x>:DELTa:Y?`

Related Commands

CALCulate:AUDio:SPECtrum:MARKer<x>:DELTa:X? (on page 353)

Arguments

None

Returns

`<NRf>` the delta marker amplitude for the selected marker.

Examples

`CALCulate:AUDio:SPECtrum:MARKer1:DELTa:Y?` might return 3.435 indicating that the delta marker amplitude is 3.435 dB.

CALCulate:AUDio:SPECtrum:MARKer<x>:MAXimum (No Query Form)

Moves the selected marker to the highest peak on the spectrum trace.

Conditions

Measurement views: Audio Spectrum

Group

Calculate commands

Syntax

`CALCulate:AUDio:SPECtrum:MARKer<x>:MAXimum`

Arguments

None

Examples

`CALCulate:AUDio:SPECtrum:MARKer1:MAXimum` moves Marker 1 (M1) to the highest peak on the trace.

CALCulate:AUDio:SPECtrum:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the selected marker to the next peak higher in amplitude on the spectrum trace.

Conditions

Measurement views: Audio Spectrum

Group

Calculate commands

Syntax

`CALCulate:AUDio:SPECtrum:MARKer<x>:PEAK:HIGHer`

Related Commands

CALCulate:AUDio:SPECtrum:MARKer<x>:PEAK:LOWer (on page 356)

Arguments

None

Examples

`CALCulate:AUDio:SPECtrum:MARKer1:PEAK:HIGHer` moves Marker 1 (M1) to the next higher peak in amplitude on the trace.

CALCulate:AUDio:SPECtrum:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the selected marker to the next peak to the left on the spectrum trace.

Conditions

Measurement views: Audio Spectrum

Group

Calculate commands

Syntax

`CALCulate:AUDio:SPECtrum:MARKer<x>:PEAK:LEFT`

Related Commands

CALCulate:AUDio:SPECtrum:MARKer<x>:PEAK:RIGHt (on page 356)

Arguments

None

Examples

`CALCulate:AUDio:SPECtrum:MARKer1:PEAK:LEFT` moves Marker 1 (M1) to the next peak to the left on the trace.

CALCulate:AUDio:SPECtrum:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the selected marker to the next peak lower in amplitude on the spectrum trace.

Conditions

Measurement views: Audio Spectrum

Group

Calculate commands

Syntax

```
CALCulate:AUDio:SPECtrum:MARKer<x>:PEAK:LOWer
```

Related Commands

CALCulate:AUDio:SPECtrum:MARKer<x>:PEAK:HIGHer (on page 355)

Arguments

None

Examples

`CALCulate:AUDio:SPECtrum:MARKer1:PEAK:LOWer` moves Marker 1 (M1) to the next lower peak in amplitude on the trace.

CALCulate:AUDio:SPECtrum:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the selected marker to the next peak to the right on the spectrum trace.

Conditions

Measurement views: Audio Spectrum

Group

Calculate commands

Syntax

```
CALCulate:AUDio:SPECtrum:MARKer<x>:PEAK:RIGHt
```

Related Commands

CALCulate:AUDio:SPECtrum:MARKer<x>:PEAK:LEFT (on page 355)

Arguments

None

Examples

`CALCulate:AUDio:SPECtrum:MARKer1:PEAK:RIGHt` moves Marker 1 (M1) to the next peak to the right on the trace.

CALCulate:AUDio:SPECtrum:MARKer<x>:TRACe

Sets or queries the trace on which the specified marker is placed in the spectrum measurement.

Conditions

Measurement views: Audio Spectrum

Group

Calculate commands

Syntax

```
CALCulate:AUDio:SPECtrum:MARKer<x>:TRACe TRACE1
CALCulate:AUDio:SPECtrum:MARKer<x>:TRACe?
```

Arguments

`TRACE1` places the specified marker on Trace 1.

Examples

`CALCulate:AUDio:SPECtrum:MARKer1 TRACE1` places Marker 1 (M1) on Trace 1.

CALCulate:AUDio:SPECtrum:MARKer<x>:X

Sets or queries the horizontal position of the selected marker on the spectrum trace.

Conditions

Measurement views: Audio Spectrum

Group

Calculate commands

Syntax

```
CALCulate:AUDio:SPECtrum:MARKer<x>:X <value>
CALCulate:AUDio:SPECtrum:MARKer<x>:X?
```

Related Commands

CALCulate:AUDio:SPECtrum:MARKer<x>:Y? (on page 357)

Arguments

`<value> ::= <NRf>` specifies the horizontal position of the marker. Range: Start to Stop frequency (left to right edge of the horizontal axis). Using an out-of-range value causes an execution error (-222, "Data out of range").

Examples

`CALCulate:AUDio:SPECtrum:MARKer1:X 800` places Marker 1 (M1) at 800 Hz on the spectrum trace.

CALCulate:AUDio:SPECtrum:MARKer<x>:Y? (Query Only)

Queries the marker amplitude of the selected marker on the spectrum trace.

Conditions

Measurement views: Audio Spectrum

Group

Calculate commands

Syntax

CALCulate:AUDio:SPECtrum:MARKer<x>:Y?

Related Commands

CALCulate:AUDio:SPECtrum:MARKer<x>:X (on page 357)

Arguments

None

Returns

<NRf> the marker amplitude for the selected marker.

Examples

CALCulate:AUDio:SPECtrum:MARKer1:Y? might return 53.818 indicating that Marker 1 (M1) is at 53.818 dBm.

CALCulate:AVTime[:MEASview<y>]:MARKer<x>:DELTa:X? (Query Only)

Returns the delta marker time for the selected marker in the Amplitude versus Time measurement.

Conditions

Measurement views: Amplitude versus Time

Group

Calculate commands

Syntax

CALCulate:AVTime[:MEASview<y>]:MARKer<x>:DELTa:X?

Related Commands

CALCulate:MARKer:ADD (on page 458)

CALCulate:AVTime[:MEASview<y>]:MARKer<x>:DELTa:Y? (on page 359)

Arguments

The parameter <x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

<NRf> Delta marker time for the selected marker.

Examples

CALCULATE:AVTIME:MEASVIEW1:MARKER1:DELTA:X? might return 38.0E-9, indicating that the delta marker time is 38.0 ns on the trace of Amplitude versus Time Measurement View 1.

CALCulate:AVTime[:MEASview<y>]:MARKer<x>:DELTa:Y? (Query Only)

Returns the delta marker amplitude for the selected marker in the Amplitude versus Time measurement.

Conditions

Measurement views: Amplitude versus Time

Group

Calculate commands

Syntax

`CALCulate:AVTime[:MEASview<y>]:MARKer<x>:DELTa:Y?`

Related Commands

CALCulate:MARKer:ADD (on page 458)

CALCulate:AVTime[:MEASview<y>]:MARKer<x>:DELTa:X? (on page 358)

Arguments

The parameter `<x>` = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter `<y>` = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

`<NRf>` Delta marker amplitude for the selected marker.

Examples

`CALCULATE:AVTIME:MEASVIEW1:MARKER1:DELTA:Y?` might return 23.45, indicating that the delta marker amplitude is 23.45 dB on the trace of Amplitude versus Time Measurement View 1.

CALCulate:AVTime[:MEASview<y>]:MARKer<x>:MAXimum (No Query Form)

Moves the selected marker to the highest peak on the trace in the Amplitude versus Time measurement.

Conditions

Measurement views: Amplitude versus Time with specified Measurement view number.

Group

Calculate commands

Syntax

`CALCulate:AVTime[:MEASview<y>]:MARKer<x>:MAXimum`

Arguments

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

CALCULATE:AVTIME:MEASVIEW1:MARKER1:MAXIMUM moves Marker 1 (M1) to the highest peak on the trace of Amplitude versus Time Measurement View 1.

CALCulate:AVTime[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the selected marker to the next peak higher in amplitude on the Amplitude versus Time trace.

Conditions

Measurement views: Amplitude versus Time with specified Measurement view number

Group

Calculate commands

Syntax

CALCulate:AVTime[:MEASview<y>]:MARKer<x>:PEAK:HIGHer

Related Commands

CALCulate:AVTime[:MEASview<y>]:MARKer<x>:PEAK:LOWer (on page 361)

Arguments

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

CALCULATE:AVTIME:MEASVIEW1:MARKER1:PEAK:HIGHER moves Marker 1 (M1) to the next peak higher in amplitude on the trace of Amplitude versus Time Measurement View 1.

CALCulate:AVTime[:MEASview<y>]:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the selected marker to the next peak to the left on the trace in the Amplitude versus Time measurement.

Conditions

Measurement views: Amplitude versus Time

Group

Calculate commands

Syntax

CALCulate:AVTime[:MEASview<y>]:MARKer<x>:PEAK:LEFT

Related Commands

CALCulate:AVTime[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (on page 361)

Arguments

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`CALCulate:AVTime:MEASVIEW1:MARKer1:PEAK:LEFT` moves Marker 1 (M1) to the next peak to the left on the trace of Amplitude versus Time Measurement View 1.

CALCulate:AVTime[:MEASview<y>]:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the selected marker to the next peak lower in amplitude on the Amplitude versus Time trace

Conditions

Measurement views: Amplitude versus Time with specified Measurement view number.

Group

Calculate commands

Syntax

`CALCulate:AVTime[:MEASview<y>]:MARKer<x>:PEAK:LOWer`

Related Commands

CALCulate:AVTime[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (on page 360)

Arguments

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`CALCULATE:AVTIME:MEASVIEW1:MARKER1:PEAK:LOWER` moves Marker 1 (M1) to the next peak lower in amplitude on the trace of Amplitude versus Time Measurement View 1.

CALCulate:AVTime[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the selected marker to the next peak to the right on the trace in the Amplitude versus Time measurement.

Conditions

Measurement views: Amplitude versus Time

Group

Calculate commands

Syntax

`CALCulate:AVTime[:MEASview<y>]:MARKer<x>:PEAK:RIGHt`

Related Commands

CALCulate:AVTime[:MEASview<y>]:MARKer<x>:PEAK:LEFT (on page 360)

Arguments

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`CALCULATE:AVTIME:MEASVIEW1:MARKER1:PEAK:RIGHT` moves Marker 1 (M1) to the next peak to the right on the trace of Amplitude versus Time Measurement View 1.

CALCulate:AVTime[:MEASview<y>]:MARKer<x>:TRACe

Sets or queries the trace on which the specified marker is placed in the Amplitude versus Time measurement.

Conditions

Measurement views: Amplitude versus Time with specified Measurement view number

Group

Calculate commands

Syntax

```
CALCulate:AVTime[:MEASview<y>]:MARKer<x>:TRACe {TRACE1 | TRACE2 | TRACE3 | TRACE4 | TRACE5 | TRACE6
    | TRACE7}
CALCulate:AVTime[:MEASview<y>]:MARKer<x>:TRACe?
```

Arguments

`TRACE1` places the selected marker on Trace 1.

`TRACE2` places the selected marker on Trace 2.

`TRACE3` places the selected marker on Trace 3.

`TRACE4` places the selected marker on Trace 4 (Math trace).

`TRACE5` places the selected marker on Trace 5 (Bitmap trace).

`TRACE6` places the selected marker on Trace 6 (Ogram).

`TRACE7` places marker on Trace 7 (Ogram line).

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1.

Examples

`CALCULATE:AVTIME:MEASVIEW1:MARKER1:TRACe TRACE1` places Marker 1 (M1) on Trace 1 of Amplitude versus Time Measurement View 1.

CALCulate:AVTime[:MEASview<y>]:MARKer<x>:X

Sets or queries the horizontal position of the selected marker in the Amplitude versus Time measurement.

Conditions

Measurement views: Amplitude versus Time

Group

Calculate commands

Syntax

```
CALCulate:AVTime[:MEASview<y>]:MARKer<x>:X <value>
CALCulate:AVTime[:MEASview<y>]:MARKer<x>:X?
```

Related Commands

CALCulate:AVTime[:MEASview<y>]:MARKer<x>:Y? (on page 363)

Arguments

<value>::=<NRf> specifies the horizontal position of the marker. Range: (analysis offset) to [(analysis offset) + (analysis length)].

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

CALCULATE:AVTIME:MEASVIEW1:MARKER1:X 1.5U places Marker 1 (M1) at 1.5 µs on the trace of Amplitude versus Time Measurement View 1.

CALCulate:AVTime[:MEASview<y>]:MARKer<x>:Y? (Query Only)

Queries the marker amplitude of the selected marker in the Amplitude versus Time measurement.

Conditions

Measurement views: Amplitude versus Time

Group

Calculate commands

Syntax

```
CALCulate:AVTime[:MEASview<y>]:MARKer<x>:Y?
```

Related Commands

CALCulate:AVTime[:MEASview<y>]:MARKer<x>:X (on page 362)

Arguments

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

<NRf> Marker amplitude of the selected marker.

Examples

CALCULATE:AVTIME:MEASVIEW1:MARKER1:Y? might return -2.73, indicating Marker 1 (M1) is at -2.73 dBm on the trace of Amplitude versus Time Measurement View 1.

CALCulate:BLUEtooth:CONSte:MARKer<x>:DELTa:X[:TIME]? (Query only)

Queries the delta marker time for the selected marker on the Bluetooth display.

The parameter <x> = 1 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Conditions

Measurement views: Bluetooth Constellation

Group

Calculate commands

Syntax

```
CALCulate:BLUEtooth:CONSte:MARKer<x>:DELTa:X[:TIME]?
```

Arguments

None

Returns

`<NRf>` Delta marker time for the selected marker.

Use `[SENSe]:BLUEtooth:ANALysis:TIMe:UNITs` to select the time unit: symbolsor seconds.

Examples

`CALCulate:BLUEtooth:CONSte:MARKer1:DELTa:X:TIME?` might return 62.75, Indicating that the delta marker time is 62.75 symbols.

CALCulate:BLUEtooth:CONSte:MARKer<x>:FDEViation? (Query Only)

Queries the frequency deviation for the selected marker in the Bluetooth Constellation measurement.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: Bluetooth Constellation.

Applicable when the Trace Type is set to Freq Dev.

Group

Calculate commands

Syntax

```
CALCulate:BLUEtooth:CONSte:MARKer<x>:FDEViation?
```

Related Commands

The following commands are applicable when the Trace Type is set to IQ:

CALCulate:BLUETooth:CONSte:MARKer<x>:MAGNitude? (on page 365)

CALCulate:BLUETooth:CONSte:MARKer<x>:PHASe? (on page 367)

Use " " to change the Trace Type.

Arguments

None

Returns

<NRf> The frequency deviation readout for the selected marker.

Examples

CALCulate:BLUEtooth:CONSte:MARKer1:FDEViation? might return 102.824E+3, indicating the frequency deviation readout of Marker 1 (M1) is 102.824 kHz.

CALCulate:BLUETooth:CONSte:MARKer<x>:MAGNitude? (Query Only)

Queries the magnitude readout of the selected marker in the Bluetooth Constellation.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: Bluetooth Constellation

Applicable when the Trace Type is set to IQ.

Group

Calculate commands

Syntax

CALCulate:BLUETooth:CONSte:MARKer<x>:MAGNitude?

Arguments

None

Returns

<NRf> The magnitude for the selected marker.

Examples

CALCulate:BLUETooth:CONSte:MARKer1:MAGNitude? might return 923E-3, indicating the magnitude readout of Marker 1 (M1) is 923E-3.

CALCulate:BLUEtooth:CONSte:MARKer<x>:MAXimum (No Query Form)

Positions the selected marker at the symbol in the center of the analyzed time record.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: Bluetooth Constellation

Group

Calculate commands

Syntax

```
CALCulate:BLUEtooth:CONSte:MARKer<x>:MAXimum
```

Arguments

None

Examples

`CALCulate:BLUEtooth:CONSte:MARKer1:MAXimum` positions Marker 1 (M1) at the symbol in the center of the time record.

CALCulate:BLUEtooth:CONSte:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the selected marker in the time domain to the next symbol number to the left, relative to the previous marker position.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: Bluetooth Constellation

Group

Calculate commands

Syntax

```
CALCulate:BLUEtooth:CONSte:MARKer<x>:PEAK:LEFT
```

Related Commands

CALCulate:BLUEtooth:CONSte:MARKer<x>:PEAK:RIGHt (on page 366)

Arguments

None

Examples

`CALCulate:BLUEtooth:CONSte:MARKer1:PEAK:LEFT` moves Marker 1 (M1) in the time domain to the next lower symbol number.

CALCulate:BLUEtooth:CONSte:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the selected marker in the time domain to the next higher symbol number, relative to the previous marker position.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: Bluetooth Constellation

Group

Calculate commands

Syntax

CALCulate:BLUEtooth:CONSte:MARKer<x>:PEAK:RIGHt

Related Commands

CALCulate:BLUEtooth:CONSte:MARKer<x>:PEAK:LEFT (on page 366)

Arguments

None

Examples

CALCulate:BLUEtooth:CONSte:MARKer1:PEAK:RIGHT moves the Marker 1 (M1) in the time domain to the next higher symbol number.

CALCulate:BLUETooth:CONSte:MARKer<x>:PHASe? (Query Only)

Queries the phase readout of the selected marker in the Bluetooth Constellation.

Conditions

Measurement views: Bluetooth Constellation.

Applicable when the Trace Type is set to IQ.

Group

Calculate commands

Syntax

CALCulate:BLUETooth:CONSte:MARKer<x>:PHASe?

Arguments

None

Returns

<NRf>

The phase readout for the selected marker in degrees between -180 and +180.

Examples

CALCulate:BLUETooth:CONSte:MARKer1:PHASe? might return 35.74, indicating the phase readout of Marker 1 (M1) is 35.74°.

CALCulate:BLUEtooth:CONSte:MARKer<x>:SYMBol? (Query Only)

Queries the symbol readout value for the selected marker in the Bluetooth Constellation measurement.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: Bluetooth Constellation

Group

Calculate commands

Syntax

```
CALCulate:BLUEtooth:CONSte:MARKer<x>:SYMBol?
```

Related Commands

CALCulate:BLUEtooth:CONSte:MARKer<x>:X (on page 369)

Arguments

None

Returns

<NRf> The symbol readout for the selected marker.

Examples

CALCulate:BLUEtooth:CONSte:MARKer1:SYMBol? might return 62.00, indicating the Symbol readout of Marker 1 (M1) is 62.

CALCulate:BLUEtooth:CONSte:MARKer<x>:Trace

Sets or queries the Trace for the selected marker on the Bluetooth Constellation display.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: Bluetooth Constellation

Group

Calculate commands

Syntax

```
CALCulate:BLUEtooth:CONSte:MARKer<x>:Trace <Enum>
CALCulate:BLUEtooth:CONSte:MARKer<x>:Trace?
```

Arguments

<Enum> ::= {TRACE1 | TRACE2}. The arguments specify which trace to place the marker on.

Examples

CALCulate:BLUEtooth:CONSte:MARKer1:TRACe TRACE1 places Marker 1 (M1) on the Trace 1.

CALCulate:BLUETooth:CONSte:MARKer<x>:VALue? (Query Only)

Queries the value readout of the selected marker in the Bluetooth Constellation.

Conditions

Measurement views: Bluetooth Constellation

Group

Calculate commands

Syntax

```
CALCulate:BLUETooth:CONSte:MARKer<x>:VALue?
```

Arguments

None

Returns

<NRf> The value readout for the selected marker.

Examples

CALCulate:BLUETooth:CONSte:MARKer2:VALue? might return 1.00, indicating the value readout of Marker 2(M2) is 1.

CALCulate:BLUEtooth:CONSte:MARKer<x>:X

Sets or queries the time position of the selected marker on the Bluetooth Constellation.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: Bluetooth Constellation

Group

Calculate commands

Syntax

```
CALCulate:BLUEtooth:CONSte:MARKer<x>:X <value>
CALCulate:BLUEtooth:CONSte:MARKer<x>:X?
```

Arguments

<value> ::= <NRf> specifies the time position of the marker. Range (analysis offset) to [(analysis offset) + (analysis length)].

Examples

CALCulate:BLUEtooth:CONSte:MARKER1:X -1.63875ms places the Marker 1 (M1) at -1.63875 ms on the CONStellation trace.

CALCulate:BLUEtooth:EDIagram:MARKer<x>:DELTa:X[:TIME]? (Query Only)

Returns the delta marker time for the selected marker on the Bluetooth eye diagram display.

The parameter <x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: Bluetooth eye diagram

Group

Calculate commands

Syntax

```
CALCulate:BLUEtooth:EDIagram:MARKer<x>:DELTa:X[:TIME]?
```

Related Commands

CALCulate:BLUEtooth:EDIagram:MARKer<x>:DELTa:Y? (on page 370)

Arguments

None

Returns

<NRf> the delta marker time for the selected marker.

Use [SENSe]:BLUEtooth:ANALysis:TIMe:UNITs to select the time unit: symbols or seconds.

Examples

CALCulate:BLUEtooth:EDIagram:MARKer1:DELTa:X:TIME? might return 62.75, indicating that the delta marker time is 62.75 symbols.

CALCulate:BLUEtooth:EDIagram:MARKer<x>:DELTa:Y? (Query Only)

Returns the delta marker amplitude for the selected markers in the Bluetooth eye diagram.

The parameter <x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: Bluetooth eye diagram

Group

Calculate commands

Syntax

CALCulate:BLUEtooth:EDIagram:MARKer<x>:DELTa:Y?

Related Commands

CALCulate:BLUEtooth:EDIagram:MARKer<x>:DELTa:X[:TIME]? (on page 369)

Returns

<NRf>

the delta marker amplitude for the selected marker. Units are in Hz if the Trace Type is set to Freq Dev and no units if the Trace Type is set to IQ.

Examples

CALCulate:BLUEtooth:EDIagram:MARKer1:DELTa:Y? might return -1.043, indicating that the delta marker amplitude is -1.043.

CALCulate:BLUEtooth:EDIagram:MARKer<x>:MAXimum (No Query Form)

Moves the selected marker to the highest peak on the Bluetooth eye diagram trace.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: Bluetooth eye diagram

Group

Calculate commands

Syntax

`CALCulate:BLUEtooth:EDIagram:MARKer<x>:MAXimum`

Arguments

None

Examples

`CALCulate:BLUEtooth:EDIagram:MARKer1:MAXImum` moves Marker 1 (M1) to the Highest peak on the trace.

CALCulate:BLUEtooth:EDIagram:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the selected marker to the next peak higher in amplitude on the Bluetooth eye diagram trace.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: Bluetooth eye diagram

Group

Calculate commands

Syntax

`CALCulate:BLUEtooth:EDIagram:MARKer<x>:PEAK:HIGHer`

Related Commands

CALCulate:BLUEtooth:EDIagram:MARKer<x>:PEAK:LOWer (on page 372)

Arguments

None

Examples

`CALCulate:BLUEtooth:EDIagram:MARKer1:PEAK:HIGHer` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

CALCulate:BLUEtooth:EDIagram:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the selected marker to the next peak to the left on the Bluetooth eye diagram trace.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: Bluetooth eye diagram

Group

Calculate commands

Syntax

CALCulate:BLUEtooth:EDIagram:MARKer<x>:PEAK:LEFT

Related Commands

CALCulate:BLUEtooth:EDIagram:MARKer<x>:PEAK:RIGHt (on page 372)

Arguments

None

Examples

CALCulate:BLUEtooth:EDIagram:MARKer1:PEAK:LEFT moves Marker 1 (M1) to the next peak to the left on the trace.

CALCulate:BLUEtooth:EDIagram:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the selected marker to the next peak lower in amplitude on the Bluetooth eye diagram trace.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

Conditions

Measurement views: Bluetooth eye diagram

Group

Calculate commands

Syntax

CALCulate:BLUEtooth:EDIagram:MARKer<x>:PEAK:LOWer

Related Commands

CALCulate:BLUEtooth:EDIagram:MARKer<x>:PEAK:HIGHer (on page 371)

Arguments

None

Examples

CALCulate:BLUEtooth:EDIagram:MARKer1:PEAK:LOWer moves Marker 1(M1) to the next peak lower in amplitude on the trace.

CALCulate:BLUEtooth:EDIagram:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the selected marker to the next peak to the right on the Bluetooth eye diagram trace.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: Bluetooth eye diagram

Group

Calculate commands

Syntax

```
CALCulate:BLUEtooth:EDIagram:MARKer<x>:PEAK:RIGHt
```

Related Commands

CALCulate:BLUEtooth:EDIagram:MARKer<x>:PEAK:LEFT (on page 371)

Arguments

None

Examples

`CALCulate:BLUEtooth:EDIagram:MARKer1:PEAK:RIGHT` moves Marker 1 (M1) to the next peak to the right on the trace.

CALCulate:BLUEtooth:EDIagram:MARKer<x>:TRACe

The command places the selected marker on the I or Q trace in the Bluetooth eye diagram display. The query returns the name of the trace where the current marker resides.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADDcommand.

Conditions

Measurement views: Bluetooth eye diagram

Group

Calculate commands

Syntax

```
CALCulate:BLUEtooth:EDIagram:MARKer<x>:TRACe <Enum>
CALCulate:BLUEtooth:EDIagram:MARKer<x>:TRACe?
```

Arguments

`<Enum> ::= {TRACE1 | TRACE2}.`

The arguments specify which trace to place the marker on.

Trace1 specifies the I trace and Trace2 specifies the Q trace.

Returns

For I and Q traces, TRACE1 identifies the I trace and TRACE2 identifies the Q trace.

Examples

`CALCulate:BLUEtooth:EDIagram:MARKer1:TRACe TRACE1` places Marker 1 (M1) on the I trace.

CALCulate:BLUEtooth:EDIagram:MARKer<x>:X[:TIME]

Sets or queries the horizontal position (time) of the selected marker in the Bluetooth eye diagram measurement.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: Bluetooth eye diagram

Group

Calculate commands

Syntax

```
CALCulate:BLUEtooth:EDIagram:MARKer<x>:X[:TIME] <value>
CALCulate:BLUEtooth:EDIagram:MARKer<x>:X[:TIME]?
```

Related Commands

CALCulate:BLUEtooth:EDIagram:MARKer<x>:Y? (on page 374)

Arguments

`<value> ::= <NRf>` the horizontal position (time) of the marker.

Use [SENSe]:BLUEtooth:ANALysis:TIMe:UNITs to select the time unit: symbols or seconds.

Examples

`CALCulate:BLUEtooth:EDIagram:MARKer1:X:TIME 38.5` places Marker 1 (M1) at 38.5 symbols on the trace.

CALCulate:BLUEtooth:EDIagram:MARKer<x>:Y? (Query Only)

Returns the vertical position of the selected marker in the Bluetooth Eye diagram measurement.

The parameter `<x>` = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: Bluetooth eye diagram

Group

Calculate commands

Syntax

```
CALCulate:BLUEtooth:EDIagram:MARKer<x>:Y?
```

Related Commands

CALCulate:BLUEtooth:EDIagram:MARKer<x>:X[:TIME] (on page 373)

Arguments

None

Returns

`<value> ::= <NRf>` the vertical position of the selected marker. The units are Hz if the Trace Type is set to Freq Dev; if the Trace Type is set to IQ, there are no units.

Examples

`CALCulate:BLUEtooth:EDIagram:MARKer1:Y?` might return `571.8E-3`, indicating Marker 1 (M1) is at 0.5718.

CALCulate:BLUEtooth:FDVTIME:MARKer<x>:DELTa:X[:TIME]? (Query Only)

Queries the delta marker time for the selected marker on the Frequency Deviation vs. Time display

The parameter <x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Use [SENSe]:BLUEtooth:ANALysis:TIMe:UNITs to select the time unit symbols or seconds.

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Calculate commands

Syntax

`CALCulate:BLUEtooth:FDVTIME:MARKer<x>:DELTa:X[:TIME]?`

Related Commands

CALCulate:BLUEtooth:FDVTIME:MARKer<x>:DELTa:Y? (on page 375)

Arguments

None

Returns

`<NRf>` the delta marker time for the selected marker.

Examples

`CALCULATE:BLUETOOTH:FDVTIME:MARKER1:DELTa:X:TIME?` might return 62.75, indicating that the delta marker time is 62.75 symbols.

CALCulate:BLUEtooth:FDVTIME:MARKer<x>:DELTa:Y? (Query Only)

Returns the delta marker amplitude for the selected marker in the Frequency Deviation vs. Time display.

The parameter <x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Calculate commands

Syntax

`CALCulate:BLUEtooth:FDVTIME:MARKer<x>:DELTa:Y?`

Related Commands

CALCulate:BLUEtooth:FDVTIME:MARKer<x>:DELTa:X[:TIME]? (on page 374)

Arguments

None

Returns

`<NRf>` the delta marker amplitude for the selected marker.

Examples

`CALCulate:BLUEtooth:FDVTIME:MARKer1:DELTa:Y?` return -1.043, indicating that the delta marker amplitude is -1.043.

CALCulate:BLUEtooth:FDVTime:MARKer<x>:MAXimum (No Query Form)

Moves the selected marker to the highest peak on the trace in the Frequency Deviation vs. Time display.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Calculate commands

Syntax

```
CALCulate:BLUEtooth:FDVTime:MARKer<x>:MAXimum
```

Arguments

None

Examples

`CALCULATE:BLUETOOTH:FDVTIME:MARKER1:MAXIMUM` moves Marker 1 (M1) to the highest peak on the trace.

CALCulate:BLUEtooth:FDVTime:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the selected marker to the next peak higher in amplitude in the Frequency Deviation vs. Time display.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Calculate commands

Syntax

```
CALCulate:BLUEtooth:FDVTime:MARKer<x>:PEAK:HIGHer
```

Related Commands

CALCulate:BLUEtooth:FDVTime:MARKer<x>:PEAK:LOWer (on page 377)

Arguments

None

Examples

`CALCULATE:BLUETOOTH:FDVTIME:MARKER1:PEAK:HIGHER` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

CALCulate:BLUEtooth:FDVTime:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the selected marker to the next peak to the left on the trace in the Frequency Deviation vs. Time display.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Calculate commands

Syntax

`CALCulate:BLUEtooth:FDVTime:MARKer<x>:PEAK:LEFT`

Related Commands

CALCulate:BLUEtooth:CONSte:MARKer<x>:PEAK:RIGHt (on page 366)

Arguments

None

Examples

`CALCULATE:BLUETOOTH:FDVTIME:MARKER1:PEAK:LEFT` moves Marker 1 (M1) to the next peak to the left on the trace.

CALCulate:BLUEtooth:FDVTime:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the selected marker to the next peak lower in amplitude on Frequency Deviation vs. Time display.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Calculate commands

Syntax

`CALCulate:BLUEtooth:FDVTime:MARKer<x>:PEAK:LOWer`

Related Commands

CALCulate:BLUEtooth:FDVTime:MARKer<x>:PEAK:HIGHer (on page 376)

Arguments

None

Examples

`CALCULATE:BLUETOOTH:FDVTIME:MARKER1:PEAK:LOWER` moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

CALCulate:BLUEtooth:FDVTime:MARKer<x>:PEAK:RIGHT (No Query Form)

Moves the selected marker to the next peak to the right on the trace in the Frequency Deviation vs. Time display.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Calculate commands

Syntax

```
CALCulate:BLUEtooth:FDVTime:MARKer<x>:PEAK:RIGHT
```

Related Commands

CALCulate:BLUEtooth:FDVTime:MARKer<x>:PEAK:LEFT (on page 377)

Arguments

None

Examples

`CALCULATE:BLUETOOTH:FDVTIME:MARKER1:PEAK:RIGHT` moves Marker 1 (M1) to the next peak to the right on the trace.

CALCulate:BLUEtooth:FDVTIME:MARKer<x>:X[:TIME]

Sets or queries the marker time in the Frequency Deviation vs. Time display.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Use [SENSe]:BLUEtooth:ANALysis:TIMe:UNITs to select the time unit symbols (default) or seconds.

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Calculate commands

Syntax

```
CALCulate:BLUEtooth:FDVTIME:MARKer<x>:X[:TIME] <value>
CALCulate:BLUEtooth:FDVTIME:MARKer<x>:X[:TIME]?
```

Related Commands

CALCulate:BLUEtooth:FDVTIME:MARKer<x>:Y (on page 379)

Arguments

`<value> ::= <NRf>` specifies the marker time in seconds or in symbols based on the Units specified under the Analysis Time.

Examples

`CALCULATE:BLUETOOTH:FDVTIME:MARKER1:X:TIME -234.5E-6` places Marker 1 (M1) at -234.5 µs on the trace.

CALCulate:BLUEtooth:FDVTIME:MARKer<x>:Y

Sets or queries the marker amplitude in the Frequency Deviation vs. Time display.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Calculate commands

Syntax

`CALCulate:BLUEtooth:FDVTIME:MARKer<x>:Y`

Related Commands

CALCulate:BLUEtooth:FDVTIME:MARKer<x>:X[:TIME] (on page 378)

Arguments

`<value> ::= <NRf>` specifies the marker frequency value in Hz.

Examples

`CALCULATE:BLUETOOTH:FDVTIME:MARKER1:Y?` might return 571.8E-3, indicating Marker 1 (M1) is 0.5718 Hz.

CALCulate:BOBW:MARKer<x>:DELTa:X? (Query Only)

Returns the delta marker frequency for the selected marker on the Bluetooth 20 dB bandwidth measurement.

The parameter <x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: BT 20 dB Bandwidth

Group

Calculate commands

Syntax

`CALCulate:BOBW:MARKer<x>:DELTa:X?`

Related Commands

CALCulate:BOBW:MARKer<x>:DELTa:Y? (on page 380)

Arguments

None

Returns

`<NRf>` Delta marker frequency for the selected marker in Hz.

Examples

CALCULATE:BOBW:MARKER1:DELTA:X? might return 62.75E+3, indicating that the delta marker time is 62.75 kHz.

CALCulate:BOBW:MARKer<x>:DELTa:Y? (Query Only)

Returns the delta marker amplitude for the selected marker on the Bluetooth 20 dB bandwidth measurement.

The parameter <x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: BT 20 dB Bandwidth

Group

Calculate commands

Syntax

CALCulate:BOBW:MARKer<x>:DELTa:Y?

Related Commands

CALCulate:BOBW:MARKer<x>:DELTa:X? (on page 379)

Arguments

None

Returns

<NRf> the Delta marker amplitude for the selected marker.

Examples

CALCULATE:BOBW:MARKER1:DELTA:Y? might return 23.45, indicating that the delta marker amplitude is 23.45 dB.

CALCulate:BOBW:MARKer<x>:MAXimum (No Query Form)

Positions the selected marker at the maximum value of the display.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: BT 20 dB Bandwidth

Group

Calculate commands

Syntax

CALCulate:BOBW:MARKer<x>:MAXimum

Arguments

None

Examples

CALCULATE:BOBW:MARKER1:MAXIMUM positions Marker 1 (M1) at the maximum value of the display.

CALCulate:BOBW:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the selected marker to the next peak higher in amplitude on the BT 20dB bandwidth trace.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: BT 20 dB Bandwidth

Group

Calculate commands

Syntax

`CALCulate:BOBW:MARKer<x>:PEAK:HIGHer`

Related Commands

CALCulate:BOBW:MARKer<x>:PEAK:LOWer (on page 381)

Arguments

None

Examples

`CALCULATE:BOBW:MARKER1:PEAK:HIGHER` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

CALCulate:BOBW:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the selected marker in the time domain to the left, relative to the previous marker position on the trace.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: BT 20 dB Bandwidth

Group

Calculate commands

Syntax

`CALCulate:BOBW:MARKer<x>:PEAK:LEFT`

Arguments

None

Examples

`CALCULATE:BOBW:MARKER1:PEAK:LEFT` moves the Marker 1 (M1) to the next peak to the left on the trace.

CALCulate:BOBW:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the selected marker to the next peak lower in amplitude on the BT 20dB bandwidth trace.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: BT 20 dB Bandwidth

Group

Calculate commands

Syntax

```
CALCulate:BOBW:MARKer<x>:PEAK:LOWer
```

Arguments

None

Examples

`CALCULATE:BOBW:MARKER1:PEAK:LOWER` moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

CALCulate:BOBW:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the selected marker in the time domain to the right, relative to the previous marker position on the trace.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: BT 20 dB Bandwidth

Group

Calculate commands

Syntax

```
CALCulate:BOBW:MARKer<x>:PEAK:RIGHt
```

Arguments

None

Examples

`CALCULATE:BOBW:MARKER1:PEAK:RIGHT` moves the Marker 1 (M1) to the next peak to the right on the trace.

CALCulate:BOBW:MARKer<x>[:SET]:CENTer (No Query Form)

Sets the center frequency to the value at the marker position in the BT 20 dB Bandwidth view.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: BT 20 dB Bandwidth

Group

Calculate commands

Syntax

```
CALCulate:BOBW:MARKer<x>[:SET]:CENTer
```

Arguments

None

Examples

CALCULATE:BOBW:MARKER1:SET:CENTER sets the center frequency to the value at the Marker 1 (M1) position.

CALCulate:BOBW:MARKer<x>:X

Sets or queries the time position of the selected marker on the BT 20 dB Bandwidth measurement.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: BT 20 dB Bandwidth

Group

Calculate commands

Syntax

```
CALCulate:BOBW:MARKer<x>:X <value>
CALCulate:BOBW:MARKer<x>:X?
```

Related Commands

CALCulate:BOBW:MARKer<x>:Y? (on page 383)

Arguments

<value> ::= <NRf> specifies the horizontal position of the marker.

Range: Start to Stop frequency (left to right edge of the horizontal axis). Using an out-of-range value causes an execution error (-222, "Data out of range").

Examples

CALCULATE:BOBW:MARKER1:X 2.4800000000E+9 places the Marker 1 (M1) at 2.4800000000E+9 on the BT 20dB Bandwidth trace.

CALCulate:BOBW:MARKer<x>:Y? (Query Only)

Returns the amplitude position of the selected marker on the Bluetooth 20 dB Bandwidth trace.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: BT 20 dB Bandwidth

Group

Calculate commands

Syntax

```
CALCulate:BOBW:MARKer<x>:Y?
```

Related Commands

CALCulate:BOBW:MARKer<x>:DELTa:X? (on page 379)

Arguments

None

Returns

<value> ::= <NRf> specifies the vertical position of the marker.

Examples

CALCULATE:BOBW:MARKER1:Y? returns the marker amplitude in dB on the BT 20dB Bandwidth trace.

CALCulate:CONSte[:MEASview<y>]:MARKer<x>:DELTa:X[:TIME]? (Query only)

Returns the delta marker time for the selected marker on the Constellation trace.

The parameter <x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Constellation

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Constellation).

Group

Calculate commands

Syntax

CALCulate:CONSte[:MEASview<y>]:MARKer<x>:DELTa:X[:TIME]?

Returns

<NRf> Delta marker time for the selected marker.

The unit can be changed by the [SENSe]:DDEMod:TIMe:UNITs command.

Examples

CALCulate:CONSte:MEASview2:MARKer1:DELTa:X:TIME? might return -62.75, indicating that the delta marker time is -62.75 symbols for measview 2 of constellation.

CALCulate:CONSte[:MEASview<y>]:MARKer<x>:FDEViation? (Query only)

Queries the frequency deviation of the selected marker in the Constellation measurement for an FSK modulated signal.

The parameter <x> = Marker 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

The command is valid for 2, 4, 8, 16 FSK or C4FM modulated signals.

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Constellation).

Group

Calculate commands

Syntax

```
CALCulate:CONSte[:MEASview<y>]:MARKer<x>:FDEViation?
```

Related Commands

CALCulate:CONSte[:MEASview<y>]:MARKer<x>:MAGNitude? (on page 385)

CALCulate:CONSte[:MEASview<y>]:MARKer<x>:PHASe? (on page 387)

Arguments

None

Returns

<NRf> the frequency deviation readout for the selected marker.

Examples

CALCulate:CONSte:MEASview2:MARKer1:FDEViation? might return 102.824E+3, indicating the frequency deviation readout of Marker 1 (M1) is 102.824 kHz for measview 2 of constellation.

CALCulate:CONSte[:MEASview<y>]:MARKer<x>:MAGNitude? (Query only)

Queries the magnitude readout of the selected marker in the Constellation measurement.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Constellation

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Constellation).

Group

Calculate commands

Syntax

```
CALCulate:CONSte[:MEASview<y>]:MARKer<x>:MAGNitude?
```

Related Commands

CALCulate:CONSte[:MEASview<y>]:MARKer<x>:X (on page 389)

Arguments

None

Returns

<NRf> the magnitude readout for the selected marker.

Examples

`CALCulate:CONSte:MEASview2:MARKer1:MAGNitude?` might return 0.713927, indicating the magnitude readout of Marker 1 (M1) is 0.713927 for measview 2 of constellation.

CALCulate:CONSte[:MEASview<y>]:MARKer<x>:MAXimum (No query form)

Positions the selected marker at the symbol in the center of the time record.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Constellation

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Constellation).

Group

Calculate commands

Syntax

`CALCulate:CONSte[:MEASview<y>]:MARKer<x>:MAXimum`

Arguments

None

Examples

`CALCulate:CONSte:MEASview2:MARKer1:MAXimum` positions Marker 1 (M1) at the symbol in the center of the time record for measview 2 of constellation.

CALCulate:CONSte[:MEASview<y>]:MARKer<x>:PEAK:LEFT (No query form)

Moves the selected marker in the time domain to the next lower symbol number, relative to the previous marker position.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Constellation

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Constellation).

Group

Calculate commands

Syntax

`CALCulate:CONSte[:MEASview<y>]:MARKer<x>:PEAK:LEFT`

Related Commands

CALCulate:CONSte[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (on page 387)

Arguments

None

Examples

`CALCulate:CONSte:MEASview2:MARKer1:PEAK:LEFT` moves Marker 1 (M1) in the time domain to the next lower symbol number for measview 2 of constellation.

CALCulate:CONSte[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (No query form)

Moves the selected marker in the time domain to the next higher symbol number, relative to the previous marker position.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Constellation

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Constellation).

Group

Calculate commands

Syntax

`CALCulate:CONSte[:MEASview<y>]:MARKer<x>:PEAK:RIGHt`

Related Commands

CALCulate:CONSte[:MEASview<y>]:MARKer<x>:PEAK:LEFT (on page 386)

Arguments

None

Examples

`CALCulate:CONSte:MEASview2:MARKer1:PEAK:RIGHT` moves the Marker 1 (M1) in the time domain to the next higher symbol number for measview 2 of constellation.

CALCulate:CONSte[:MEASview<y>]:MARKer<x>:PHASe? (Query only)

Queries the phase readout of the selected marker in the Constellation measurement.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Constellation

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Constellation).

Group

Calculate commands

Syntax

`CALCulate:CONSte[:MEASview<y>]:MARKer<x>:PHASe?`

Related Commands

CALCulate:CONSte[:MEASview<y>]:MARKer<x>:X (on page 389)

Arguments

None

Returns

`<NRf>` The phase readout for the selected marker.

Examples

`CALCulate:CONSte:MEASview2:MARKer1:PHASe` might return 35.74, indicating the phase readout of Marker 1 (M1) is 35.74 ° for measview 2 of constellation.

CALCulate:CONSte[:MEASview<y>]:MARKer<x>:SYMBol? (Query only)

Queries the symbol readout of the selected marker in the Constellation measurement.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Constellation

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Constellation).

Group

Calculate commands

Syntax

`CALCulate:CONSte[:MEASview<y>]:MARKer<x>:SYMBol?`

Related Commands

CALCulate:CONSte[:MEASview<y>]:MARKer<x>:X (on page 389)

Arguments

None

Returns

`<NRf>` the symbol readout for the selected marker.

Examples

CALCulate:CONSte:MEASview2:MARKer1:SYMBol? might return 62.00, indicating the symbol readout of Marker 1 (M1) is 62 for measview 2 of constellation.

CALCulate:CONSte[:MEASview<y>]:MARKer<x>:VALue? (Query only)

Queries the value readout of the selected marker in the Constellation measurement.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Constellation

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Constellation).

Group

Calculate commands

Syntax

CALCulate:CONSte[:MEASview<y>]:MARKer<x>:VALue?

Related Commands

CALCulate:CONSte[:MEASview<y>]:MARKer<x>:X (on page 389)

Arguments

None

Returns

<NRf> the value readout for the selected marker.

Examples

CALCulate:CONSte:MEASview2:MARKer1:VALue? might return 2.00, indicating the value readout of Marker 1 (M1) is 2 for measview 2 of constellation.

CALCulate:CONSte[:MEASview<y>]:MARKer<x>:X

Sets or queries the time position of the selected marker on the Constellation trace.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Constellation

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Constellation).

Group

Calculate commands

Syntax

```
CALCulate:CONSte[:MEASview<y>]:MARKer<x>:X<value>
CALCulate:CONSte[:MEASview<y>]:MARKer<x>:X?
```

Related Commands

CALCulate:CONSte[:MEASview<y>]:MARKer<x>:MAGNitude? (on page 385)

CALCulate:CONSte[:MEASview<y>]: MARKer<x>:PHASe? (on page 387)

CALCulate:CONSte[:MEASview<y>]:MARKer<x>:SYMBol? (on page 388)

Arguments

`<value>::=<NRf>` specifies the time position of the marker. Range: (analysis offset) to [(analysis offset) + (analysis length)].

Examples

`CALCulate:CONSte:MEASview2:MARKer1:X` -1.63875 m places the Marker 1 (M1) at -1.63875 ms on the Constellation trace for measview 2 of constellation.

CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:DELTa:X[:TIME]? (Query only)

Returns the delta marker time for the selected marker on the Demodulated I&Q versus Time trace.

The parameter <x> = 1 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Demodulated I&Q versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod I&Q versus Time).

Group

Calculate commands

Syntax

```
CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:DELTa:X[:TIME]?
```

Related Command

CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:DELTa:Y? (on page 391)

Arguments

None

Returns

`<NRf>` Delta marker time for the selected marker.

Use the `[SENSe]:DDEMod:TIMe:UNITs` command to select the time unit, symbols (default) or seconds.

Examples

CALCulate:DIQVtime:MEASview1:MARKer1:DELTa:X:TIME? might return 62.75, indicating that the delta marker time is 62.75 symbols measurement view 1.

CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:DELTa:Y? (Query only)

Returns the delta marker amplitude for the selected marker on the Demodulated I&Q versus Time trace.

The parameter <x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Demodulated I&Q versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod I&Q versus Time).

Group

Calculate commands

Syntax

CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:DELTa:Y?

Related Commands

CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:DELTa:X[:TIME]? (on page 390)

Arguments

None

Returns

<NRf> Delta marker amplitude for the selected marker in volts.

Examples

CALCulate:DIQVtime:MEASview1:MARKer1:DELTa:Y? might return -78.24E-3, indicating that the delta marker amplitude is -78.24 mV measurement view 1.

CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:MAXimum (No query form)

Moves the selected marker to the highest peak on the Demodulated I&Q versus Time trace.

The parameter <x> = Marker 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Demodulated I&Q versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod I&Q versus Time).

Group

Calculate commands

Syntax

`CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:MAXimum`

Arguments

None

Examples

`CALCulate:DIQVtime:MEASview1:MARKer1:MAXimum` moves Marker 1 (M1) to the highest peak on the trace measurement view 1.

CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (No query form)

Moves the selected marker to the next peak higher in amplitude on the Demodulated I&Q versus Time trace.

The parameter <x> = Marker 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Demodulated I&Q versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod I&Q versus Time).

Group

Calculate commands

Syntax

`CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:PEAK:HIGHer`

Related Commands

CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:PEAK:LOWer (on page 393)

Arguments

None

Examples

`CALCulate:DIQVtime:MEASview1:MARKer1:PEAK:HIGHer` moves Marker 1 (M1) to the next peak higher in amplitude on the trace measurement view 1.

CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:PEAK:LEFT (No query form)

Moves the selected marker to the next peak to the left on the Demodulated I&Q versus Time trace.

The parameter <x> = Marker 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Demodulated I&Q versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod I&Q versus Time).

Group

Calculate commands

Syntax

`CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:PEAK:LEFT`

Related Commands

CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (on page 394)

Arguments

None

Examples

`CALCulate:DIQVtime:MEASview1:MARKer1:PEAK:LEFT` moves Marker 1 (M1) to the next peak to the left on the trace measurement view 1.

CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:PEAK:LOWer (No query form)

Moves the selected marker to the next peak lower in amplitude on the Demodulated I&Q versus Time trace.

The parameter <x> = Marker 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Demodulated I&Q versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod I&Q versus Time).

Group

Calculate commands

Syntax

`CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:PEAK:LOWer`

Related Commands

CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (on page 392)

Arguments

None

Examples

`CALCulate:DIQVtime:MEASview1:MARKer1:PEAK:LOWer` moves Marker 1 (M1) to the next peak lower in amplitude on the trace measurement view 1.

CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (No query form)

Moves the selected marker to the next peak to the right on the Demodulated I&Q versus Time trace.

The parameter <x> = Marker 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Demodulated I&Q versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod I&Q versus Time).

Group

Calculate commands

Syntax

`CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:PEAK:RIGHt`

Related Commands

CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:PEAK:LEFT (on page 393)

Arguments

None

Examples

`CALCulate:DIQVtime:MEASview1:MARKer1:PEAK:RIGHt` moves Marker 1 (M1) to the next peak to the right on the trace measurement view 1.

CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:TRACe

Places the selected marker on the Demodulated I&Q versus Time trace. The query returns the name of the trace on which the marker resides.

The parameter <x> = Marker 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Demodulated I&Q versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod I&Q versus Time).

Group

Calculate commands

Syntax

```
CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:TRACe <Enum>
CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:TRACe?
```

Related Commands

CALCulate:IQVTime:MARKer<x>:TRACe (on page 443)

Arguments

`<Enum>::= TRACE1 | TRACE2` specifies which trace to place the marker on. TRACE1 specifies the “I” trace and TRACE2 specifies the “Q” trace.

Returns

For I and Q traces, TRACE1 identifies the “I” trace and TRACE2 identifies the “Q” trace.

Examples

`CALCulate:DIQVtime:MEASview1:MARKer1:TRACe TRACE1` places Marker 1 (M1) on the I trace measurement view 1.

CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:X[:TIME]

Sets or queries the horizontal position (time) of the selected marker in the Demodulated I&Q versus Time measurement.

The parameter <x> = Marker 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Demodulated I&Q versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod I&Q versus Time).

Group

Calculate commands

Syntax

```
CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:X[:TIME] <value>
CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:X[:TIME]?
```

Related Commands

CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:Y? (on page 396)

Arguments

<value>::=<NRf> specifies the horizontal position (time) of the marker. Use the [SENSe]:DDEMod:TIMe:UNITs command to select the time unit, symbols (default) or seconds.

Examples

CALCulate:DIQVtime:MEASview1:MARKER1:X:TIME 38.5 places Marker 1 (M1) at 38.5 symbols on the trace measurement view 1.

CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:Y? (Query only)

Queries the marker amplitude of the selected marker in the Demod I&Q versus Time measurement.

The parameter <x> = Marker 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Demodulated I&Q versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod I&Q versus Time).

Group

Calculate commands

Syntax

CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:Y?

Related Commands

CALCulate:DIQVtime[:MEASview<y>]:MARKer<x>:X[:TIME] (on page 395)

Arguments

None

Returns

<NRf> Marker amplitude of the selected marker in volts.

Examples

CALCulate:DIQVtime:MEASview1:MARKer1:Y? might return 25.803E-3, indicating Marker 1 (M1) is at 25.803 mV measurement view 1.

CALCulate:DPX:MARKer<x>:DELTa:X:AMPLitude? (Query Only)

Returns the delta marker amplitude for the selected marker on the DPX spectrum trace.

The parameter <x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command and attached to the Bitmap trace using the CALCulate:DPX:MARKer<x>:TRACe command.

Conditions

Measurement views: DPX spectrum

Group

Calculate commands

Syntax

```
CALCulate:DPX:MARKer<x>:DELTa:X:AMPLitude?
```

Related Commands

CALCulate:DPX:MARKer<x>:X[:FREQuency] (on page 403), CALCulate:DPX:MARKer<x>:Y? (on page 405)

Arguments

None

Returns

<NRf> Delta marker amplitude for the selected marker in dB.

Examples

CALCULATE:DPX:MARKER1:DELTA:X:AMPLITUDE? might return 35.12, indicating that the delta marker amplitude is 35.12 dB for Marker 1 (M1).

CALCulate:DPX:MARKer<x>:DELTa:X[:FREQuency]? (Query Only)

Returns the delta marker frequency for the selected marker on the DPX spectrum trace.

The parameter <x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

Conditions

Measurement views: DPX spectrum

Group

Calculate commands

Syntax

```
CALCulate:DPX:MARKer<x>:DELTa:X[:FREQuency]?
```

Related Commands

CALCulate:DPX:MARKer<x>:DELTa:X:AMPLitude? (on page 396)

CALCulate:DPX:MARKer<x>:DELTa:Y? (on page 397)

Arguments

None

Returns

<NRf> Delta marker frequency for the selected marker in Hz.

Examples

CALCULATE:DPX:MARKER1:DELTA:X? might return 1.28E+6, indicating that the delta marker frequency is 1.28 MHz for Marker 1 (M1).

CALCulate:DPX:MARKer<x>:DELTa:Y? (Query Only)

Returns the delta marker amplitude for the selected marker on the DPX spectrum trace.

The parameter <x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

Conditions

Measurement views: DPX spectrum

Group

Calculate commands

Syntax

CALCulate:DPX:MARKer<x>:DELTa:Y?

Related Commands

CALCulate:DPX:MARKer<x>:DELTa:X[:FREQuency]? (on page 397)

Arguments

None

Returns

<NRf> Delta marker amplitude for the selected marker.

Examples

CALCULATE:DPX:MARKER1:DELTA:Y? might return 23.45, indicating that the delta marker amplitude is 23.45 dB.

CALCulate:DPX:MARKer<x>:MAXimum (No Query Form)

Moves the selected marker to the highest peak on the DPX spectrum trace.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

Conditions

Measurement views: DPX spectrum

Group

Calculate commands

Syntax

CALCulate:DPX:MARKer<x>:MAXimum

Arguments

None

Examples

CALCULATE:DPX:MARKER1:MAXIMUM moves Marker 1 (M1) to the highest peak on the trace.

CALCulate:DPX:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the selected marker to the next peak higher in amplitude on the DPX spectrum trace.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

Conditions

Measurement views: DPX spectrum

Group

Calculate commands

Syntax

`CALCulate:DPX:MARKer<x>:PEAK:HIGHer`

Related Commands

CALCulate:DPX:MARKer<x>:PEAK:LOWer (on page 399)

Arguments

None

Examples

`CALCULATE:DPX:MARKER1:PEAK:HIGHER` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

CALCulate:DPX:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the selected marker to the next peak to the left on the DPX spectrum trace.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

Conditions

Measurement views: DPX spectrum

Group

Calculate commands

Syntax

`CALCulate:DPX:MARKer<x>:PEAK:LEFT`

Related Commands

CALCulate:DPX:MARKer<x>:PEAK:RIGHt (on page 400)

Arguments

None

Examples

`CALCULATE:DPX:MARKER1:PEAK:LEFT` moves Marker 1 (M1) to the next peak to the left on the trace.

CALCulate:DPX:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the selected marker to the next peak lower in amplitude on the DPX spectrum trace.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

Conditions

Measurement views: DPX spectrum

Group

Calculate commands

Syntax

```
CALCulate:DPX:MARKer<x>:PEAK:LOWer
```

Related Commands

CALCulate:DPX:MARKer<x>:PEAK:HIGHer (on page 398)

Arguments

None

Examples

`CALCULATE:DPX:MARKER1:PEAK:LOWER` moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

CALCulate:DPX:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the selected marker to the next peak to the right on the DPX spectrum trace.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

Conditions

Measurement views: DPX spectrum

Group

Calculate commands

Syntax

```
CALCulate:DPX:MARKer<x>:PEAK:RIGHt
```

Related Commands

CALCulate:DPX:MARKer<x>:PEAK:LEFT (on page 399)

Arguments

None

Examples

`CALCULATE:DPX:MARKER1:PEAK:RIGHT` moves Marker 1 (M1) to the next peak to the right on the trace.

CALCulate:DPX:MARKer<x>:POWer:DENSity? (Query Only)

Returns the power density of the specified marker in the DPX measurement.

Conditions

Measurement view: DPX

The parameter <x> = 0 to 4.

Group

Calculate commands

Syntax

```
CALCulate:DPX:MARKer<x>:POWer:DENSity?
```

Returns

<NRf> is the power density of the selected marker in dBm/Hz.

Examples

CALCULATE:DPX:MARKER1:POWER:DENSITY? might return 80.2432546111, indicating the power density is 80.24 dBm/Hz.

CALCulate:DPX:MARKer<x>:POWer:IDensity? (Query Only)

Returns the integrated power density reading between the MR (reference marker) and the selected marker in the DPX measurement.

Conditions

Measurement view: DPX

The parameter <x> = 0 to 4.

Group

Calculate commands

Syntax

CALCulate:DPX:MARKer<x>:POWer:IDensity?

Returns

<NRf> is the integrated power density reading between the MR (reference marker) and the selected marker in dBm.

Examples

CALCULATE:DPSA:MARKER1:POWER:REFDENSITY? might return -78.56 dBm/Hz.

CALCulate:DPX:MARKer<x>:POWer:INTEgrated? (Query Only)

Returns the integrated power reading between the MR (reference marker) and the selected marker in the DPX measurement.

Conditions

Measurement view: DPX

The parameter <x> = 0 to 4.

Group

Calculate commands

Syntax

CALCulate:DPX:MARKer<x>:POWer:INTEgrated?

Returns

<NRf> is the integrated power reading between MR (reference marker) and the selected marker in dBm.

Examples

CALCULATE:DPX:MARKER1:POWER:INTEGRATED? might return -14.66, indicating the integrated power is -14.66 dBm.

CALCulate:DPX:MARKer<x>[:SET]:CENTer (No Query Form)

Sets the center frequency to the marker frequency in the DPX spectrum view.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

Conditions

Measurement views: DPX spectrum

Group

Calculate commands

Syntax

```
CALCulate:DPX:MARKer<x>[:SET]:CENTer
```

Arguments

None

Examples

`CALCULATE:DPX:MARKER1:SET:CENTER` sets the center frequency to the marker frequency in the DPX spectrum view.

CALCulate:DPX:MARKer<x>:TRACe

Sets or queries the trace to attach the specified marker to in the DPX spectrum measurement.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

Conditions

Measurement views: DPX spectrum

Group

Calculate commands

Syntax

```
CALCulate:DPX:MARKer<x>:TRACe { BITMAP | TRACE1 | TRACE2 |TRACE3 | TRACE4 }
CALCulate:DPX:MARKer<x>:TRACe?
```

Arguments

`BITMAP` attaches the specified marker to the Bitmap trace.

`TRACE1` attaches the specified marker to the +Peak trace.

`TRACE2` attaches the specified marker to the -Peak trace.

`TRACE3` attaches the specified marker to the Average trace.

`TRACE4` attaches the specified marker to the Math trace.

Examples

`CALCULATE:DPX:MARKER1:TRACE TRACE1` attaches Marker 1 (M1) to the +Peak trace.

CALCulate:DPX:MARKer<x>:X:AMPLitude

Sets or queries the amplitude position of the selected marker in the DPX spectrum view. This command is valid for the marker on the bitmap trace (refer to the CALCulate:DPX:MARKer<x>:TRACe command). The frequency position is set by the CALCulate:DPX:MARKer<x>:X[:FREQuency] command.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

Conditions

Measurement views: DPX spectrum

Group

Calculate commands

Syntax

```
CALCulate:DPX:MARKer<x>:X:AMPLitude <value>
CALCulate:DPX:MARKer<x>:X:AMPLitude?
```

Arguments

`<value>::=<NRf>` specifies the amplitude position of the marker.Range: -100 to 0 dBm.

Examples

`CALCULATE:DPX:MARKER1:X:AMPLITUDE -34.5dBm` places Marker 1 (M1) at -34.5 dBm.

CALCulate:DPX:MARKer<x>:X[:FREQuency]

Sets or queries the frequency position of the selected marker in the DPX spectrum view.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

The command is only valid for the Spectrum and Frequency plots.

Conditions

Measurement views: DPX spectrum

Group

Calculate commands

Syntax

```
CALCulate:DPX:MARKer<x>:X[:FREQuency] <value>
CALCulate:DPX:MARKer<x>:X[:FREQuency]?
```

Related Commands

CALCulate:DPX:MARKer<x>:Y? (on page 405)

Arguments

`<value>::=<NRf>` specifies the frequency position of the marker.Range: Start to Stop frequency (left to right edge of the horizontal axis).Using an out-of-range value causes an execution error (-222, "Data out of range").

Examples

`CALCULATE:DPX:MARKER1:X:FREQUENCY 800MHz` places Marker 1 (M1) at 800 MHz on the trace.

CALCulate:DPX:MARKer<x>:X:PHASe

Sets or queries the phase position of the selected marker in the DPX phase plot.

The parameter <x> = Marker 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Conditions

Measurement views: DPX spectrum

Group

Calculate commands

Syntax

```
CALCulate:DPX:MARKer<x>:X:PHASe <value>
CALCulate:DPX:MARKer<x>:X:PHASe?
```

Arguments

`<value>::<NRf>` phase setting in degrees for the specified marker.

Examples

`CALCulate:DPX:MARKer0:X:PHASe?` might return 94.0000000 indicating the phase for the reference marker (M0) is 94.00°.

CALCulate:DPX:MARKer<x>:X:TIME

Sets or queries the time position of the selected marker in the DPX spectrum view.

The parameter <x> = Marker 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

The command is valid for the Amplitude, Frequency, and Phase plots.

Conditions

Measurement views: DPX spectrum

Group

Calculate commands

Syntax

```
CALCulate:DPX:MARKer<x>:X:TIME <value>
CALCulate:DPX:MARKer<x>:X:TIME?
```

Arguments

`<value>::<NRf>` Time position of the specified marker.

Examples

`CALCulate:DPX:MARKer0:X:TIME?` might return 1.000000000E–3 indicating the time position of the reference marker (MR0) is 1.000 ms.

CALCulate:DPX:MARKer<x>:Y? (Query Only)

Queries the vertical position of the selected marker in the DPX spectrum view. The data occurrence rate is returned for the bitmap trace, and the amplitude value for the +peak, -peak, average, and math traces. The horizontal position can be set by the CALCulate:DPX:MARKer<x>:X:AMPLitude and CALCulate:DPX:MARKer<x>:X[:FREQuency] commands.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

Conditions

Measurement views: DPX spectrum

Group

Calculate commands

Syntax

```
CALCulate:DPX:MARKer<x>:Y?
```

Arguments

None

Returns

`<NRf>` The value type depends on which trace the marker is placed on (refer to the CALCulate:DPX:MARKer<x>:TRACe command):

The amplitude value is returned in dBm for the marker on the +peak, -peak, average, or math trace. The data occurence rate is returned in percent (%) for the marker on the bitmap trace.

Examples

`CALCULATE:DPX:MARKER1:Y?` might return -34.28 indicating Marker 1 (M1) is at -34.28 dBm when it is placed on the +peak, -peak, average, or math trace.

`CALCULATE:DPX:MARKER1:Y?` might return 76.5 indicating Marker 1 (M1) is at 76.5% when it is placed on the bitmap trace.

CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:DELTa:X[:TIME]? (Query only)

Returns the delta marker time for the selected marker on the eye diagram trace.

The parameter <x> = 1 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Eye diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of the measurement (Example: Eye Diagram).

Group

Calculate commands

Syntax

```
CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:DELTa:X[:TIME]?
```

Related Commands

CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:DELTa:Y? (on page 406)

Arguments

None

Returns

<NRf> Delta marker time for the selected marker. Use the [SENSe]:DDEMod:TIMe:UNITs command to select the time unit:symbols (default) or seconds.

Examples

CALCulate:EDIagram:MEASview2:MARKer1:DELTa:X:TIME? might return 62.75, indicating that the delta marker time is 62.75 symbols for measview2 of the Eye diagram.

CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:DELTa:Y? (Query only)

Returns the delta marker amplitude for the selected marker on the eye diagram trace.

The parameter <x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Eye diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Eye diagram).

Group

Calculate commands

Syntax

CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:DELTa:Y?

Related Commands

CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:DELTa:X[:TIME]? (on page 405)

Arguments

None

Returns

<NRf> Delta marker amplitude for the selected marker.

Examples

CALCulate:EDIagram:MEASview2:MARKer<x>:DELTa:Y? might returns -1.043, indicating that the delta marker amplitude is -1.043 for measview2 of the Eye diagram.

CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:MAXimum (No query form)

Moves the selected marker to the highest peak on the eye diagram trace.

The parameter <x> = Marker 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Eye diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Eye Diagram).

Group

Calculate commands

Syntax

```
CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:MAXimum
```

Arguments

None

Examples

`CALCulate:EDIagram:MEASview1:MARKer<x>:MAXimum` moves the Marker 1 (M1) to the highest peak on the trace for measview1 of the Eye diagram.

CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (No query form)

Moves the selected marker to the next peak higher in amplitude on the eye diagram trace.

The parameter <x> = Marker 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Eye diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Eye Diagram).

Group

Calculate commands

Syntax

```
CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:PEAK:HIGHer
```

Related Commands

CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:PEAK:LOWer (on page 408)

Arguments

None

Examples

`CALCulate:EDIagram:MEASview1:MARKer<x>:PEAK:HIGHer` moves the Marker 1 (M1) to the next peak higher in amplitude on the trace for measview2 of the Eye diagram.

CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:PEAK:LEFT (No query form)

Moves the selected marker to the next peak to the left on the eye diagram trace.

The parameter <x> = Marker 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Eye diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Eye diagram).

Group

Calculate commands

Syntax

`CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:PEAK:LEFT`

Related Commands

CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (on page 409)

Arguments

None

Examples

`CALCulate:EDIagram:MEASview2:MARKer<x>:PEAK:LEFT` moves the Marker 1 (M1) to the next peak to the left on the trace for measview2 of the Eye diagram.

CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:PEAK:LOWer (No query form)

Moves the selected marker to the next peak lower in amplitude on the eye diagram trace.

The parameter <x> = Marker 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Eye diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Eye Diagram).

Group

Calculate commands

Syntax

`CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:PEAK:LOWer`

Related Commands

CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (on page 407)

Arguments

None

Examples

`CALCulate:EDIagram:MEASview2:MARKer<x>:PEAK:LOWer` moves the Marker 1 (M1) to the next peak lower in amplitude on the trace for measview2 of the Eye diagram.

CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (No query form)

Moves the selected marker to the next peak to the right on the eye diagram trace.

The parameter <x> = Marker 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Eye diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Eye diagram).

Group

Calculate commands

Syntax

`CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:PEAK:RIGHt`

Related Commands

CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:PEAK:LEFT (on page 408)

Arguments

None

Examples

`CALCulate:EDIagram:MEASview2:MARKer<x>:PEAK:RIGHt` moves the Marker 1 (M1) to the next peak to the right on the trace for measview2 of the Eye diagram.

CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:TRACe

Places the selected marker on the I or Q trace in the eye diagram display. The query returns the name of the trace on which the marker resides. It is valid for all types of modulation except 2|4|8|16 FSK and C4FM.

The parameter <x> = Marker 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Eye diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Eye diagram).

Group

Calculate commands

Syntax

```
CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:TRACe<Enum>
CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:TRACe?
```

Related Commands

CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:X[:TIME] (on page 410)

Arguments

`<Enum>::= TRACE1 | TRACE2` specifies which trace to place the marker on.TRACE1 specifies the “I” trace and TRACE2 specifies the “Q” trace.

Returns

For I and Q traces, TRACE1 identifies the “I” trace and TRACE2 identifies the “Q” trace.

Examples

`CALCulate:EDIagram:MEASview2:MARKer1:TRACe TRACE1` places the Marker 1 (M1) on the I trace in an eye diagram for the measview2.

CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:X[:TIME]

Sets or queries the horizontal position (time) of the selected marker in the eye diagram measurement.

The parameter <x> = Marker 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Eye diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Eye Diagram).

Group

Calculate commands

Syntax

```
CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:X[:TIME]<value>
CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:X[:TIME]?
```

Related Commands

CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:Y? (on page 411)

Arguments

`<value>::=<NRf>` specifies the horizontal position (time) of the marker. Use the `[SENSe]:DDEMod:TIMe:UNITs` command to select the time unit:symbols (default) or seconds.

Examples

`CALCulate:EDIagram:MEASVIEW2:MARKer1:X:TIME` 38.5 places Marker 1 (M1) at 38.5 symbols on the trace for the measview2 of Eye diagram.

CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:Y? (Query only)

Queries the vertical position of the selected marker in the eye diagram measurement.

The parameter <x> = Marker 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Eye diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Eye diagram).

Group

Calculate commands

Syntax

```
CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:Y?
```

Related Commands

CALCulate:EDIagram[:MEASview<y>]:MARKer<x>:X[:TIME] (on page 410)

Arguments

None

Returns

`<NRf>` Vertical position of the selected marker.

Examples

`CALCulate:EDIagram:MEASview2:MARKer<x>:Y?` might return 571.8E-3, indicating Marker 1 (M1) is at 0.5718 for the measview 2 of Eye diagram.

CALCulate:EMC:MARKer<x>:DELTa:X? (Query Only)

Returns the delta marker frequency value for the selected marker in the EMC view.

<x> = 1,2,3,4 is the marker.

Conditions

Measurement view: EMC - EMI Display

Group

Calculate commands

Syntax

`CALCulate:EMC:MARKer<x>:DELTa:X?`

Arguments

None

Returns

`<NRf>`

Examples

`CALCulate:EMC:MARKer1:DELTa:X?` might return -178.9375, indicating that the delta marker frequency of Marker 1 (M1) is -178.93 KHz.

CALCulate:EMC:MARKer<x>:DELTa:Y? (Query Only)

Returns the delta marker amplitude for the selected marker in the EMC view.

<x> = 1,2,3,4 is the marker.

Conditions

Measurement view: EMC - EMI Display

Group

Calculate commands

Syntax

`CALCulate:EMC:MARKer<x>:DELTa:Y?`

Arguments

None

Returns

`<NRf>`

Examples

`CALCulate:EMC:MARKer1:DELTa:Y?` might return -2.2589, indicating that the delta marker Amplitude of Marker 1 (M1) is -2.25 dB.

CALCulate:EMC:MARKer<x>:MAXimum (No Query Form)

Positions the specified marker at the maximum value of the display on the EMC trace.

<x> = 0,1,2,3,4 is the marker

Conditions

Measurement view: EMC - EMI Display

Group

Calculate commands

Syntax

`CALCulate:EMC:MARKer<x>:MAXimum`

Arguments

None

Returns

None

Examples

`CALCulate:EMC:MARKer1:MAXimum` positions the marker 1 (M1) at the maximum value of the display.

CALCulate:EMC:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the specified marker to the next peak higher in amplitude on the EMC trace.

<x> = 0,1,2,3,4 is the marker

Conditions

Measurement view: EMC - EMI Display

Group

Calculate commands

Syntax

`CALCulate:EMC:MARKer<x>:PEAK:HIGHer`

Arguments

None

Returns

None

Examples

`CALCulate:EMC:MARKer2:PEAK:HIGHer` moves Marker 2 (M2) to the next peak higher in amplitude on the trace.

CALCulate:EMC:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the specified marker to the next peak to the left on the EMC trace.

<x> = 0,1,2,3,4 is the marker

Conditions

Measurement view: EMC - EMI Display

Group

Calculate commands

Syntax

`CALCulate:EMC:MARKer<x>:PEAK:LEFT`

Arguments

None

Returns

None

Examples

`CALCulate:EMC:MARKer1:PEAK:LEFT` moves Marker 1 (M1) to the next peak to the left on the trace.

CALCulate:EMC:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the specified marker to the next peak lower in amplitude on the EMC trace.

<x> = 0,1,2,3,4 is the marker

Conditions

Measurement view: EMC - EMI Display

Group

Calculate commands

Syntax

`CALCulate:EMC:MARKer<x>:PEAK:LOWer`

Arguments

None

Returns

None

Examples

`CALCulate:EMC:MARKer2:PEAK:LOWer` moves Marker 2 (M2) to the next peak lower in amplitude on the trace.

CALCulate:EMC:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the specified marker to the next peak to the right on the EMC trace.

<x> = 0,1,2,3,4 is the marker

Conditions

Measurement view: EMC - EMI Display

Group

Calculate commands

Syntax

`CALCulate:EMC:MARKer<x>:PEAK:RIGHt`

Arguments

None

Returns

None

Examples

`CALCulate:EMC:MARKer1:PEAK:RIGHt` moves Marker 1 (M1) to the next peak to the right on the trace.

CALCulate:EMC:MARKer<x>[:SET]:CENTer (No Query Form)

Sets the center frequency to the value at the specified marker position in the EMC view.

<x> = 0,1,2,3,4 is the marker

Conditions

Measurement view: EMC - EMI Display

Group

Calculate commands

Syntax

`CALCulate:EMC:MARKer<x>[:SET]:CENTer`

Arguments

None

Returns

None

Examples

`CALCulate:EMC:MARKer1[:SET]:CENTer` sets the center frequency to the value at the Marker 1 (M1) position.

CALCulate:EMC:MARKer<x>:X

Sets or queries the frequency position of the selected marker on the EMC view.

<x> = 0,1,2,3,4 is the marker

Conditions

Measurement view: EMC - EMI Display

Group

Calculate commands

Syntax

```
CALCulate:EMC:MARKer<x>:X <value>
CALCulate:EMC:MARKer<x>:X ?
```

Arguments

`<value> = <NRf>` specifies the horizontal position of the marker.

Returns

`<NRf>`

Examples

`CALCulate:EMC:MARKer1:X` might return 254.562500000E+3, indicating that the frequency position of the Marker 1 (M1) is 254.56 KHz.

CALCulate:EMC:MARKer<x>:Y

Sets or queries the amplitude position of the selected marker in the EMC view.

<x> = 0,1,2,3,4 is the marker

Conditions

Measurement view: EMC - EMI Display

Group

Calculate commands

Syntax

```
CALCulate:EMC:MARKer<x>:Y <value>
CALCulate:EMC:MARKer<x>:Y?
```

Arguments

`<value> = <NRf>` specifies the amplitude position of the marker.

Returns

`<NRf>`

Examples

`CALCulate:EMC:MARKer1:Y` might return 8.9157, indicating that the amplitude position of Marker 1 (M1) is 8.92 dBuV.

CALCulate:EVM[:MEASview<y>]:MARKer<x>:DELTa:X? (Query only)

Returns the delta marker time for the selected marker in the EVM versus Time measurement.

The parameter <x> = 1 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: EVM versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod EVM versus Time).

Group

Calculate commands

Syntax

`CALCulate:EVM[:MEASview<y>]:MARKer<x>:DELTa:X?`

Related Commands

CALCulate:EVM[:MEASview<y>]:MARKer<x>:DELTa:Y? (on page 417)

Arguments

None

Returns

`<NRf>` Delta marker amplitude for the selected marker in percent (%).

Examples

`CALCulate:EVM:MEASview2:MARKer<x>:DELTa:X?` might return 1.62, indicating that the delta marker duration is 1.62 for measview2 of EVM.

CALCulate:EVM[:MEASview<y>]:MARKer<x>:DELTa:Y? (Query only)

Returns the delta marker amplitude for the selected marker in the EVM versus Time measurement.

The parameter <x> = 1 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD command`.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: EVM versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod EVM versus Time).

Group

Calculate commands

Syntax

`CALCulate:EVM[:MEASview<y>]:MARKer<x>:DELTa:Y?`

Related Commands

CALCulate:EVM[:MEASview<y>]:MARKer<x>:DELTa:X? (on page 416)

Arguments

None

Returns

`<NRf>` Delta marker amplitude for the selected marker in percent (%).

Examples

`CALCulate:EVM:MEASview2:MARKer<x>:DELTa:Y?` might return 1.62, indicating that the delta marker amplitude is 1.62% for measview2 of EVM.

CALCulate:EVM[:MEASview<y>]:MARKer<x>:MAXimum (No query form)

Moves the selected marker to the highest peak on the trace in the EVM versus Time measurement.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: EVM versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod EVM versus Time).

Group

Calculate commands

Syntax

`CALCulate:EVM[:MEASview<y>]:MARKer<x>:MAXimum`

Arguments

None

Examples

`CALCulate:EVM:MEASview2:MARKer<x>:MAXimum` moves Marker 1 (M1) to the highest peak on the trace for measview2 of EVM.

CALCulate:EVM[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (No query form)

Moves the selected marker to the next peak higher in amplitude on the EVM versus Time trace.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: EVM versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod EVM versus Time).

Group

Calculate commands

Syntax

`CALCulate:EVM[:MEASview<y>]:MARKer<x>:PEAK:HIGHer`

Related Commands

CALCulate:EVM[:MEASview<y>]:MARKer<x>:PEAK:LOWer (on page 419)

Arguments

None

Examples

`CALCulate:EVM:MEASview2:MARKer<x>:PEAK:HIGHer` moves Marker 1 (M1) to the next peak higher in amplitude on the trace for measview2 of EVM.

CALCulate:EVM[:MEASview<y>]:MARKer<x>:PEAK:LEFT (No query form)

Moves the selected marker to the next peak to the left on the trace in the EVM versus Time measurement.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: EVM versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod EVM versus Time).

Group

Calculate commands

Syntax

`CALCulate:EVM[:MEASview<y>]:MARKer<x>:PEAK:LEFT`

Related Commands

CALCulate:EVM[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (on page 420)

Arguments

None

Examples

`CALCulate:EVM:MEASview2:MARKer<x>:PEAK:LEFT` moves the Marker 1 (M1) to next peak to the left on the trace for measview2 of EVM.

CALCulate:EVM[:MEASview<y>]:MARKer<x>:PEAK:LOWer (No query form)

Moves the selected marker to the next peak lower in amplitude on the EVM versus Time trace.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: EVM versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod EVM versus Time)

Group

Calculate commands

Syntax

`CALCulate:EVM[:MEASview<y>]:MARKer<x>:PEAK:LOWer`

Related Commands

CALCulate:EVM[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (on page 418)

Arguments

None

Examples

`CALCulate:EVM:MEASview2:MARKer<x>:PEAK:LOWer` moves Marker 1 (M1) to the next peak lower in amplitude on the trace for measview2 of EVM.

CALCulate:EVM[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (No query form)

Moves the selected marker to the next peak to the right on the trace in the EVM versus Time measurement.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: EVM versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod EVM versus Time).

Group

Calculate commands

Syntax

`CALCulate:EVM[:MEASview<y>]:MARKer<x>:PEAK:RIGHt`

Related Commands

CALCulate:EVM[:MEASview<y>]:MARKer<x>:PEAK:LEFT (on page 419)

Arguments

None

Examples

`CALCulate:EVM:MEASview2:MARKer<x>:PEAK:RIGHt` moves the Marker 1 (M1) to the next peak to the right on the trace for measview2 of EVM.

CALCulate:EVM[:MEASview<y>]:MARKer<x>:X

Sets or queries the horizontal position of the selected marker in the EVM versus Time measurement.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: EVM versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod EVM versus Time).

Group

Calculate commands

Syntax

```
CALCulate:EVM[:MEASview<y>]:MARKer<x>:X <value>
CALCulate:EVM[:MEASview<y>]:MARKer<x>:X?
```

Arguments

`<value>::=<NRf>` specifies the horizontal position of the marker Range: (analysis offset) to (analysis offset) + (analysis length). The unit can be changed by the `[SENSe]:DDEMod:TIMe:UNITs` command.

Examples

`CALCulate:EVM:MEASview2:MARKer1:X 1.5 µs` places Marker 1 (M1) at 1.5 µs on the trace for measview 2 of EVM.

CALCulate:EVM[:MEASview<y>]:MARKer<x>:Y? (Query only)

Queries the marker amplitude of the selected marker in the EVM versus Time measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: EVM versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod EVM versus Time).

Group

Calculate commands

Syntax

```
CALCulate:EVM[:MEASview<y>]:MARKer<x>:Y?
```

Related Commands

CALCulate:EVM[:MEASview<y>]:MARKer<x>:X (on page 421)

Arguments

None

Returns

<NRf> Marker amplitude of the selected marker in percent (%).

Examples

CALCulate:EVM:MEASview2:MARKer<x>:Y? might return 5.34, indicating Marker 1 (M1) is at 5.34% for measview2 of EVM.

CALCulate:FDVTime[:MEASview<y>]:MARKer<x>:DELTa:X[:TIME]? (Query only)

Returns the delta marker time for the selected marker on the Frequency deviation versus Time trace.

The parameter <x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Frequency deviation versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Frequency deviation versus Time).

Group

Calculate commands

Syntax

CALCulate:FDVTime:MEASview<y>:MARKer<x>:DELTa:X[:TIME]?

Related Commands

CALCulate:FDVTime[:MEASVIEW<y>]:MARKer<x>:DELTa:Y? (on page 422)

Arguments

None

Returns

<NRf> Delta marker amplitude for the selected marker.

Examples

CALCULATE:FDVTime:MEASview2:MARKER1:DELTA:X? might return 120.0E-9, indicating that the delta marker duration is 120 ns.

CALCulate:FDVTime[:MEASview<y>]:MARKer<x>:DELTa:Y? (Query only)

Returns the delta marker amplitude for the selected marker on the Frequency deviation versus Time trace.

The parameter <x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Frequency deviation versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Frequency deviation versus Time).

Group

Calculate commands

Syntax

`CALCulate:FDVTime:MEASview<y>2:MARKer<x>:DELTa:Y?`

Related Commands

CALCulate:FDVTime:MEASVIEW2:MARKer<x>:DELTa:X[:TIME]? (on page 422)

Arguments

None

Returns

`<NRf>` Delta marker amplitude for the selected marker in Hz.

Examples

`CALCULATE:FDVTime:MEASview2:MARKER1:DELTA:Y?` might return 27.05E+3, indicating that the delta marker amplitude is 27.05 KHz.

CALCulate:FDVTime[:MEASview<y>]:MARKer<x>:MAXimum (No query form)

Moves the selected marker to the highest peak on the trace in the Frequency deviation versus Time measurement.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Frequency deviation vs Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Frequency deviation versus Time).

Group

Calculate commands

Syntax

`CALCulate:FDVTime:MEASview<y>:MARKer<x>:MAXimum`

Arguments

None

Examples

`CALCULATE:FDVTime:MEASview2:MARKER1:MAXIMUM` moves Marker 1 (M1) to the highest peak on the trace.

CALCulate:FDVTime[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (No query form)

Moves the selected marker to the next peak higher in amplitude on the Frequency deviation versus Time trace.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Frequency Deviation versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Frequency deviation versus Time).

Group

Calculate commands

Syntax

```
CALCulate:FDVTime[:MEASview<y>]:MARKer<x>:PEAK:HIGHer
```

Related Commands

CALCulate:FDVTime[:MEASview<y>]:MARKer<x>:PEAK:LOWer (on page 425)

Arguments

None

Examples

`CALCULATE:FDVTime:MEASview2:MARKER1:PEAK:HIGHER` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

CALCulate:FDVTime[:MEASview<y>]:MARKer<x>:PEAK:LEFT (No query form)

Moves the selected marker to the next peak to the left on the Frequency deviation versus Time trace.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Frequency deviation versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Frequency deviation versus Time).

Group

Calculate commands

Syntax

```
CALCulate:FDVTime:MEASview<y>:MARKer<x>:PEAK:LEFT
```

Related Commands

CALCulate:FDVTime[:MEASVIEW<y>]:MARKer<x>:PEAK:RIGHt (on page 425)

Arguments

None

Examples

`CALCULATE:FDVTime:MEASview2:MARKER1:PEAK:LEFT` moves Marker 1 (M1) to the next peak to the left on the trace.

CALCulate:FDVTime[:MEASview<y>]:MARKer<x>:PEAK:LOWer (No query form)

Moves the selected marker to the next peak lower in amplitude on the Frequency deviation vs Time trace.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Frequency deviation vs Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Frequency deviation versus Time).

Group

Calculate commands

Syntax

`CALCulate:FDVTime:MEASview<y>:MARKer<x>:PEAK:LOWer`

Related Commands

CALCulate:FDVTime[:MEASVIEW<y>]:MARKer<x>:PEAK:HIGHer (on page 424)

Arguments

None

Examples

`CALCULATE:FDVTime:MEASview2:MARKER1:PEAK:LOWER` moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

CALCulate:FDVTime[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (No query form)

Moves the selected marker to the next peak to the right on the trace in the Frequency deviation vs Time measurement.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Frequency deviation vs Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Frequency deviation versus Time).

Group

Calculate commands

Syntax

```
CALCulate:FDVTime:MEASview<y>:MARKer<x>:PEAK:RIGHt
```

Related Commands

CALCulate:FDVTime[:MEASVIEW<y>]:MARKer<x>:PEAK:LEFT (on page 424)

Arguments

None

Examples

`CALCULATE:FDVTime:MEASview2:MARKER1:PEAK:RIGHT` moves Marker 1 (M1) to the next peak to the right on the trace.

CALCulate:FDVTime[:MEASview<y>]:MARKer<x>:X[:TIME]

Sets or queries the horizontal position (time) of the selected marker in the Frequency deviation vs Time measurement.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Frequency deviation vs Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Frequency deviation versus Time).

Group

Calculate commands

Syntax

```
CALCulate:FDVTime:MEASview<y>:MARKer<x>:X[:TIME]<value>
CALCulate:FDVTime:MEASview<y>:MARKer<x>:X[:TIME]?
```

Related Commands

CALCulate:FDVTime[:MEASview<y>]:MARKer<x>:Y? (Query only) (on page 427)

Arguments

None.

Returns

`<NRf>` Marker amplitude of the selected marker in Hz.

Examples

`CALCULATE:FDVTime:MEASview2:MARKER1:X -15.34E+06` places Marker 1 (M1) at -15.34 MHz on the trace.

CALCulate:FDVTime[:MEASview<y>]:MARKer<x>:Y? (Query only)

Queries the marker amplitude of the selected marker in the Frequency deviation versus Time measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Frequency deviation versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Frequency deviation versus Time).

Group

Calculate commands

Syntax

`CALCulate:FDVTime:MEASview<y>:MARKer<x>:Y?`

Related Commands

CALCulate:FDVTime[:MEASVIEW<y>]:MARKer<x>:X[:TIME] (on page 426)

Arguments

None

Returns

`<NRf>` Marker amplitude of the selected marker in Hz.

Examples

`CALCULATE:FDVTime:MEASview2:MARKER1:Y?` might return -15.34E+6, indicating Marker 1 (M1) is at -15.34 MHz.

CALCulate:{FSETtling|PSETtling}:MARKer<x>:DELTa:X? (Query only)

Returns the delta marker time for the selected marker in the Frequency and Phase Settling displays.

The parameter <x> = 1 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Conditions

Measurement views: Frequency and Phase Settling

Group

Calculate commands

Syntax

`CALCulate:{FSETtling|PSETtling}:MARKer<x>:DELTa:X?`

Related Commands

CALCulate:FVTime:MARKer<x>:DELTa:Y? (on page 433)

Arguments

None

Returns

<NRf> Delta marker time for the selected marker.

Examples

CALCULATE:FSETTLING:MARKER1:DELTA:X? might return 120.0E-9,indicating that the delta marker time is 120 ns.

CALCulate:{FSETtling|PSETtling}:MARKer<x>:DELTa:Y? (Query only)

Returns the delta marker frequency for the selected marker in the Frequency and Phase Settling displays.

The parameter <x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: Frequency and Phase Settling

Group

Calculate commands

Syntax

CALCulate:{FSETtling|PSETtling}:MARKer<x>:DELTa:Y?

Related Commands

CALCulate:{FSETtling|PSETtling}:MARKer<x>:DELTa:X? (on page 427)

Arguments

None

Returns

<NRf> Delta marker frequency for the selected marker in Hz.

Examples

CALCULATE:FSETTLING:MARKER1:DELTA:Y? might return 27.05E+3, indicating that the delta marker frequency is 27.05 kHz.

CALCulate:{FSETtling|PSETtling}:MARKer<x>:MAXimum (No Query Form)

Moves the selected marker to the highest peak on the trace in the Frequency or Phase Settling display.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

Conditions

Measurement views: Frequency and Phase Settling

Group

Calculate commands

Syntax

CALCulate:{FSETtling|PSETtling}:MARKer<x>:MAXimum

Arguments

None

Examples

`CALCULATE:FSETTLING:MARKER1:MAXIMUM` moves Marker 1 (M1) to the highest peak on the trace.

CALCulate:{FSETtling|PSETtling}:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the selected marker to the next peak higher in amplitude on the Frequency or Phase Settling trace.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

Conditions

Measurement views: Frequency and Phase Settling

Group

Calculate commands

Syntax

`CALCulate:{FSETtling|PSETtling}:MARKer<x>:PEAK:HIGHer`

Related Commands

CALCulate:{FSETtling|PSETtling}:MARKer<x>:PEAK:LOWer (on page 430)

Arguments

None

Examples

`CALCULATE:FSETTLING:MARKER1:PEAK:HIGHER` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

CALCulate:{FSETtling|PSETtling}:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the selected marker to the next peak to the left on the trace in the Frequency or Phase Settling display.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

Conditions

Measurement views: Frequency and Phase Settling

Group

Calculate commands

Syntax

`CALCulate:{FSETtling|PSETtling}:MARKer<x>:PEAK:LEFT`

Related Commands

CALCulate:{FSETtling|PSETtling}:MARKer<x>:PEAK:RIGHt (on page 430)

Arguments

None

Examples

`CALCULATE:FSETTLING:MARKER1:PEAK:LEFT` moves Marker 1 (M1) to the next peak to the left on the trace.

CALCulate:{FSETtling|PSETtling}:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the selected marker to the next peak lower in amplitude on the Frequency or Phase Settling trace.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

Conditions

Measurement views: Frequency and Phase Settling

Group

Calculate commands

Syntax

`CALCulate:{FSETtling|PSETtling}:MARKer<x>:PEAK:LOWer`

Related Commands

CALCulate:{FSETtling|PSETtling}:MARKer<x>:PEAK:HIGHer (on page 429)

Arguments

None

Examples

`CALCULATE:FSETTLING:MARKER1:PEAK:LOWER` moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

CALCulate:{FSETtling|PSETtling}:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the selected marker to the next peak to the right on the trace in the Frequency or Phase Settling display.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

Conditions

Measurement views: Frequency versus Time

Group

Calculate commands

Syntax

`CALCulate:{FSETtling|PSETtling}:MARKer<x>:PEAK:RIGHt`

Related Commands

CALCulate:FVTime[:MEASview<y>]:MARKer<x>:PEAK:LEFT (on page 434)

Arguments

None

Examples

`CALCULATE:FSETTLING:MARKER1:PEAK:RIGHT` moves Marker 1 (M1) to the next peak to the right on the trace.

CALCulate:{FSETtling|PSETtling}:MARKer<x>:TRACe

Sets or queries the trace on which the specified marker is placed in the spectrum measurement.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

Conditions

Measurement views: Frequency and Phase Settling

Group

Calculate commands

Syntax

```
CALCulate:{FSETtling|PSETtling}:MARKer<x>:TRACe { TRACE1 |TRACE2 }
CALCulate:{FSETtling|PSETtling}:MARKer<x>:TRACe?
```

Arguments

`TRACE1` places the specified marker on Trace 1.

`TRACE2` places the specified marker on Trace 2.

Examples

`CALCULATE:FSETTLING:MARKER1:TRACE TRACE1` places Marker 1 (M1) on Trace 1.

CALCulate:{FSETtling|PSETtling}:MARKer<x>:X

Sets or queries the horizontal position of the selected marker in the Frequency and Phase Settling display.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

Conditions

Measurement views: Frequency and Phase Settling

Group

Calculate commands

Syntax

```
CALCulate:{FSETtling|PSETtling}:MARKer<x>:X <value>
CALCulate:{FSETtling|PSETtling}:MARKer<x>:X?
```

Related Commands

CALCulate:{FSETtling|PSETtling}:MARKer<x>:Y? (on page 432)

Arguments

`<value>::=<NRf>` specifies the horizontal position of the marker.Range: (analysis offset) to [(analysis offset) + (analysis length)].

Examples

CALCULATE:FVTIME:MARKER1:X 1.5u places Marker 1 (M1) at 1.5 µs on the trace.

CALCulate:{FSETtling|PSETtling}:MARKer<x>:Y? (Query Only)

Queries the marker amplitude of the selected marker in the Frequency and Phase Settling display.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

Conditions

Measurement views: Frequency and Phase Settling

Group

Calculate commands

Syntax

CALCulate:{FSETtling|PSETtling}:MARKer<x>:Y?

Related Commands

CALCulate:{FSETtling|PSETtling}:MARKer<x>:X (on page 431)

Arguments

None

Returns

<NRf> Marker amplitude of the selected marker in Hz or degrees.

Examples

CALCULATE:FSETTLING:MARKER1:Y? might return -15.34E+6, indicating Marker 1 (M1) is at -15.34 MHz.

CALCulate:FVTime[:MEASview<y>]:MARKer<x>:DELTa:X? (Query Only)

Returns the delta marker time for the selected marker in the Frequency versus Time measurement.

Conditions

Measurement views: Frequency versus Time

Group

Calculate commands

Syntax

CALCulate:FVTime[:MEASview<y>]:MARKer<x>:DELTa:X?

Related Commands

CALCulate:FVTime[:MEASview<y>]:MARKer<x>:DELTa:Y? (on page 433)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Returns

<NRf> Delta marker time for the selected marker.

Examples

CALCULATE:FVTIME:MEASview1:MARKER1:DELTA:X? might return 120.0E-9, indicating that the delta marker time is 120 ns.

CALCulate:FVTime[:MEASview<y>]:MARKer<x>:DELTa:Y? (Query Only)

Returns the delta marker frequency for the selected marker in the Frequency versus Time measurement.

Conditions

Measurement views: Frequency versus Time

Group

Calculate commands

Syntax

CALCulate:FVTime[:MEASview<y>]:MARKer<x>:DELTa:Y?

Related Commands

CALCulate:FVTime[:MEASview<y>]:MARKer<x>:DELTa:X? (on page 432)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Returns

<NRf> Delta marker frequency for the selected marker in Hz.

Examples

CALCULATE:FVTIME:MEASview1:MARKER1:DELTA:Y? might return 27.05E+3, indicating that the delta marker frequency is 27.05 kHz.

CALCulate:FVTime[:MEASview<y>]:MARKer<x>:MAXimum (No Query Form)

Moves the selected marker to the highest peak on the trace in the Frequency versus Time measurement.

Conditions

Measurement views: Frequency versus Time

Group

Calculate commands

Syntax

```
CALCulate:FVTime[:MEASview<y>]:MARKer<x>:MAXimum
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Examples

`CALCULATE:FVTIME:MEASview1:MARKER1:MAXIMUM` moves Marker 1 (M1) to the highest peak on the trace.

CALCulate:FVTime[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the selected marker to the next peak higher in amplitude on the Frequency versus Time trace.

Conditions

Measurement views: Frequency versus Time

Group

Calculate commands

Syntax

```
CALCulate:FVTime[:MEASview<y>]:MARKer<x>:PEAK:HIGHer
```

Related Commands

CALCulate:FVTime[:MEASview<y>]:MARKer<x>:PEAK:LOWer (on page 435)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Examples

`CALCULATE:FVTIME:MEASview1:MARKER1:PEAK:HIGHER` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

CALCulate:FVTime[:MEASview<y>]:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the selected marker to the next peak to the left on the trace in the Frequency versus Time measurement.

Conditions

Measurement views: Frequency versus Time

Group

Calculate commands

Syntax

CALCulate:FVTime[:MEASview<y>]:MARKer<x>:PEAK:LEFT

Related Commands

CALCulate:FVTime[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (on page 435)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Examples

CALCULATE:FVTIME:MEASview1:MARKER1:PEAK:LEFT moves Marker 1 (M1) to the next peak to the left on the trace.

CALCulate:FVTime[:MEASview<y>]:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the selected marker to the next peak lower in amplitude on the Frequency versus Time trace.

Conditions

Measurement views: Frequency versus Time

Group

Calculate commands

Syntax

CALCulate:FVTime[:MEASview<y>]:MARKer<x>:PEAK:LOWer

Related Commands

CALCulate:FVTime[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (on page 434)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Examples

CALCULATE:FVTIME:MEASview1:MARKER1:PEAK:LOWER moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

CALCulate:FVTime[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the selected marker to the next peak to the right on the trace in the Frequency versus Time measurement.

Conditions

Measurement views: Frequency versus Time

Group

Calculate commands

Syntax

```
CALCulate:FVTime[:MEASview<y>]:MARKer<x>:PEAK:RIGHt
```

Related Commands

CALCulate:FVTime[:MEASview<y>]:MARKer<x>:PEAK:LEFT (on page 434)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Examples

`CALCULATE:FVTIME:MEASview1:MARKER1:PEAK:RIGHT` moves Marker 1 (M1) to the next peak to the right on the trace.

CALCulate:FVTime[:MEASview<y>]:MARKer<x>:TRACe

Sets or queries the trace on which the specified marker is placed in the Frequency versus Time measurement.

Conditions

Measurement views: Frequency versus Time with specified Measurement view number.

Group

Calculate commands

Syntax

```
CALCulate:FVTime[:MEASview<y>]:MARKer<x>:TRACe {TRACE1 | TRACE5 | TRACE6 | TRACE7}
```

```
CALCulate:FVTime[:MEASview<y>]:MARKer<x>:TRACe?
```

Arguments

`TRACE1` places the selected marker on Trace 1.

`TRACE5` places the selected marker on Trace 5 (Bitmap trace).

`TRACE6` places the selected marker on Trace 6 (Ogram).

`TRACE7` places marker on Trace 7 (Ogram line).

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`CALCULATE:FVTIME:MEASVIEW1:MARKER1:TRACe TRACE1` places Marker 1 (M1) on Trace 1 of Frequency versus Time Measurement View 1.

CALCulate:FVTime[:MEASview<y>]:MARKer<x>:X

Sets or queries the horizontal position of the selected marker in the Frequency versus Time measurement.

Conditions

Measurement views: Frequency versus Time

Group

Calculate commands

Syntax

```
CALCulate:FVTime[:MEASview<y>]:MARKer<x>:X <value>
CALCulate:FVTime[:MEASview<y>]:MARKer<x>:X?
```

Related Commands

CALCulate:FVTime[:MEASview<y>]:MARKer<x>:Y? (on page 437)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

<value>::=<NRf> specifies the horizontal position of the marker. Range: (analysis offset) to [(analysis offset) + (analysis length)].

Examples

`CALCULATE:FVTIME:MEASview1:MARKER1:X 1.5u` places Marker 1 (M1) at 1.5 µs on the trace.

CALCulate:FVTime[:MEASview<y>]:MARKer<x>:Y? (Query Only)

Queries the marker amplitude of the selected marker in the Frequency versus Time measurement.

Conditions

Measurement views: Frequency versus Time

Group

Calculate commands

Syntax

```
CALCulate:FVTime[:MEASview<y>]:MARKer<x>:Y?
```

Related Commands

CALCulate:FVTime[:MEASview<y>]:MARKer<x>:X (on page 436)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Returns

<NRf> Marker amplitude of the selected marker in Hz.

Examples

`CALCULATE:FVTIME:MEASview1:MARKER1:Y?` might return -15.34E+6, indicating Marker 1 (M1) is at -15.34 MHz.

CALCulate:HARMonic:APPLy (No Query Form)

Based on the fundamental frequency and harmonic factor values the harmonic markers are applied and markers are placed in their respective positions on the plot.

Conditions

Measurement view: EMC - EMI Display

Group

Calculate commands

Syntax

```
CALCulate:HARMonic:APPLy
```

Arguments

None

Returns

None

Examples

`CALCULATE:HARMONIC:APPLY` places the harmonic markers in their respective positions.

CALCulate:HARMonic:MARKer<x>:FACTor

Sets or queries the factor value for the specified harmonic marker in EMC view.

<x> = <NR1> 1 to 5 (marker number)

Conditions

Measurement view: EMC - EMI Display

Group

Calculate commands

Syntax

```
CALCulate:HARMonic:MARKer<x>:FACTor <value>
CALCulate:HARMonic:MARKer<x>:FACTor?
```

Arguments

<value> =

Either

`OEIG` = 1/8

`QUAR` = 1/4

`HALF` = 1/2

`ONE` = 1

`TWO` = 2

`THREE` = 3

`FOUR` = 4

FIVE = 5

SIX = 6

SEVEN = 7

EIGHT = 8

NINE = 9

TEN = 10

Returns

<String> See Arguments

Examples

CALCulate:HARMonic:MARKer1:FACTor? might return 4 indicating that the factor value of harmonic marker 1 is 4.

CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:DELTa:X? (Query Only)

Returns the delta marker time for the selected marker in the RF I&Q versus Time measurement.

Conditions

Measurement views: RF I&Q versus Time

Group

Calculate commands

Syntax

CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:DELTa:X?

Related Commands

CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:DELTa:Y? (on page 439)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Returns

<NRf> Delta marker time for the selected marker.

Examples

CALCULATE:IQVTIME:MEASview1:MARKER1:DELTA:X? might return 120.0E-9, indicating that the delta marker time is 120 ns.

CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:DELTa:Y? (Query Only)

Returns the delta marker amplitude for the selected marker in the RF I&Q versus Time measurement.

Conditions

Measurement views: RF I&Q versus Time

Group

Calculate commands

Syntax

Syntax

`CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:DELTa:Y?`

Related Commands

CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:DELTa:X? (on page 439)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Returns

`<NRf>` Delta marker amplitude for the selected marker in volts.

Examples

`CALCULATE:IQVTIME:MEASview1:MARKER1:DELTA:Y?` might return -3.45E-3, indicating that the delta marker amplitude is -3.45 mV.

CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:MAXimum (No Query Form)

Moves the selected marker to the highest peak on the trace in the RF I&Q versus Time measurement.

Conditions

Measurement views: RF I&Q versus Time

Group

Calculate commands

Syntax

`CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:MAXimum`

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Examples

`CALCULATE:IQVTIME:MEASview1:MARKER1:MAXIMUM` moves Marker 1 (M1) to the highest peak on the trace.

CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the selected marker to the next peak higher in amplitude on the RF I&Q versus Time trace.

Conditions

Measurement views: RF I&Q versus Time

Group

Calculate commands

Syntax

```
CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:PEAK:HIGHer
```

Related Commands

CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:PEAK:LOWer (on page 442)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Examples

`CALCULATE:IQVTIME:MEASview1:MARKER1:PEAK:HIGHER` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the selected marker to the next peak to the left on the trace in the RF I&Q versus Time measurement.

Conditions

Measurement views: RF I&Q versus Time

Group

Calculate commands

Syntax

```
CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:PEAK:LEFT
```

Related Commands

CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (on page 442)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Examples

`CALCULATE:IQVTIME:MEASview1:MARKER1:PEAK:LEFT` moves Marker 1 (M1) to the next peak to the left on the trace.

CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the selected marker to the next peak lower in amplitude on the RF I&Q versus Time trace.

Conditions

Measurement views: RF I&Q versus Time

Group

Calculate commands

Syntax

`CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:PEAK:LOWer`

Related Commands

CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (on page 441)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Examples

`CALCULATE:IQVTIME:MEASview1:MARKER1:PEAK:LOWER` moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the selected marker to the next peak to the right on the trace in the RF I&Q versus Time measurement.

Conditions

Measurement views: RF I&Q versus Time

Group

Calculate commands

Syntax

`CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:PEAK:RIGHt`

Related Commands

CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:PEAK:LEFT (on page 441)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Examples

`CALCULATE:IQVTIME:MEASview1:MARKER1:PEAK:RIGHT` moves Marker 1 (M1) to the next peak to the right on the trace.

CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:TRACe

Places the selected marker on the I or Q trace in the RF I&Q versus Time measurement. The query command returns which trace the selected marker is placed on.

Conditions

Measurement views: RF I&Q versus Time

Group

Calculate commands

Syntax

```
CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:TRACe { TRACE1 | TRACE2 }
CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:TRACe?
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

`TRACE1` places the selected marker on the I trace.

`TRACE2` places the selected marker on the Q trace.

Examples

`CALCULATE:IQVTIME:MEASview1:MARKER1:TRACE TRACE1` places Marker 1 (M1) on the I trace.

CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:X

Sets or queries the horizontal position of the selected marker in the RF I&Q versus Time measurement.

Conditions

Measurement views: RF I&Q versus Time

Group

Calculate commands

Syntax

```
CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:X <value>
CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:X?
```

Related Commands

CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:Y? (on page 444)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

`<value>::=<NRf>` specifies the horizontal position of the marker.Range: (analysis offset) to [(analysis offset) + (analysis length)].

Examples

`CALCULATE:IQVTIME:MEASview1:MARKER1:X 1.5us` places Marker 1 (M1) at 1.5 µs on the trace.

CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:Y? (Query Only)

Queries the marker amplitude of the selected marker in the RF I&Q versus Time measurement.

Conditions

Measurement views: RF I&Q versus Time

Group

Calculate commands

Syntax

`CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:Y?`

Related Commands

CALCulate:IQVTime[:MEASview<y>]:MARKer<x>:X (on page 443)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Returns

`<NRf>` Marker amplitude of the selected marker in volts.

Examples

`CALCULATE:IQVTIME:MEASview1:MARKER1:Y?` might return 25.803E-3, indicating Marker 1 (M1) is at 25.803 mV.

CALCulate:LTE:ACLR:MARKer:<x>PEAK:HIGHer (No Query Form)

Moves the specified marker to the next peak higher in amplitude on the trace in the LTE ACLR display.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

Conditions

Measurement view: LTE ACLR

Group

Calculate commands

Syntax

`CALCulate:LTE:ACLR:MARKer:<x>PEAK:HIGHer`

Arguments

None

Examples

`CALCULATE:LTE:ACLR:MARKER1:PEAK:HIGHER` moves marker 1 (M1) to the next peak higher in amplitude on the trace.

CALCulate:LTE:ACLR:MARKer<x>:DELTa:X? (Query Only)

Returns the frequency of the specified delta marker on the trace in the LTE ACLR display. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement view: LTE ACLR

<x> parameters: <x> = 1,2,3,4 is the marker.

Group

Calculate commands

Syntax

`CALCulate:LTE:ACLR:MARKer<x>:DELTa:X?`

ReturnsCALCulate:P25:PVTime:MARKer<x>:DELTa:Y[:TIME]? (Query Only)

`<NR3>`

Examples

`CALCULATE:LTE:ACLR:MARKER1:DELTA:X?` might return 1.25000000E+6,indicating that the frequency of the specified delta marker is 1.25 MHz.

CALCulate:LTE:ACLR:MARKer<x>:DELTa:Y? (Query Only)

Returns the amplitude for the specified delta marker on the trace in the LTE ACLR display. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement view: LTE ACLR

<x> = 1,2,3,4 is the marker.

Group

Calculate commands

Syntax

CALCulate:LTE:ACLR:MARKer<x>:DELTa:Y? (on page 445)

Returns

`<NRf>`

Examples

CALC:LTEACLR:MARK1:DELT:Y? might return 23.45, indicating that the amplitude of the specified delta marker is 23.45 dB.

CALCulate:LTE:ACLR:MARKer<x>:MAXimum (No Query Form)

Positions the specified marker at the highest peak on the trace in the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

<x> parameters: 0 to 4

Group

Calculate commands

Syntax

CALCulate:LTE:ACLR:MARKer<x>:MAXimum

Arguments

None

Examples

CALCULATE:LTE:ACLR:MARKER1:MAXIMUM moves marker 1 (M1) to the highest peak on the trace.

CALCulate:LTE:ACLR:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the selected marker to the next peak to the left on the trace in the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

<x> parameters: 0 to 4

Group

Calculate commands

Syntax

CALCulate:LTE:ACLR:MARKer<x>:PEAK:LEFT

Arguments

None

Examples

CALCULATE:LTE:ACLR:MARKER1:PEAK:LEFT moves marker 1 (M1) to the next peak to the left on the trace.

CALCulate:LTE:ACLR:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the selected marker to the next peak lower in amplitude on the trace in the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

<x> parameters: 0 to 4

Group

Calculate commands

Syntax

`CALCulate:LTE:ACLR:MARKer<x>:PEAK:LOWer`

Arguments

None

Examples

`CALCULATE:LTE:ACLR:MARKER1:PEAK:LOWER` moves marker 1 (M1) to the next peak lower in amplitude on the trace.

CALCulate:LTE:ACLR:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the selected marker to the next peak to the right on the trace in the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

<x> parameters: 0 to 4

Group

Calculate commands

Syntax

`CALCulate:LTE:ACLR:MARKer<x>:PEAK:RIGHt`

Arguments

None

Examples

`CALCULATE:LTE:ACLR:MARKER1:PEAK:RIGHT` moves marker 1 (M1) to the next peak to the right on the trace.

CALCulate:LTE:ACLR:MARKer<x>:SET:CENTer (No Query Form)

Moves the specified marker to the center frequency in the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

<x> parameters: 0 to 4

Group

Calculate commands

Syntax

`CALCulate:LTE:ACLR:MARKer<x>:SET:CENTer`

Arguments

None

Examples

CALCULATE:LTE:ACLR:MARKER1:SET:CENTER moves marker 1 (M1) to the center frequency on the trace.

CALCulate:LTE:ACLR:MARKer<x>:X

Sets or queries the horizontal position of the selected marker on the trace in the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

<x> parameters: 0 to 4

Group

Calculate commands

Syntax

```
CALCulate:LTE:ACLR:MARKer<x>:X <value>
CALCulate:LTE:ACLR:MARKer<x>:X?
```

Arguments

<value>::=<NRf> specifies the horizontal position of the marker.

Returns

<NRf>

Examples

CALCULATE:LTE:ACLR:MARKER1:X 800.000000000E+6 places marker 1 (M1) at 800 MHz on the trace.

CALCulate:LTE:ACLR:MARKer<x>:Y? (Query Only)

Queries the amplitude of the selected marker on the trace in the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

<x> parameters: 0 to 4

Group

Calculate commands

Syntax

```
CALCulate:LTE:ACLR:MARKer<x>:Y?
```

Returns

<NRf>

Examples

CALCULATE:LTE:ACLR:MARKER1:Y? might return –34.28, indicating that the amplitude of marker 1 (M1) is –34.28 dBm.

CALCulate:LTE:CHSPectrum:MARKer<x>:DELTa:X? (Query Only)

Returns the frequency of the specified delta marker on the trace in the LTE Channel Spectrum display. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement view: LTE Channel Spectrum

<x> = 1,2,3,4 is the marker.

Group

Calculate commands

Syntax

`CALCulate:LTE:CHSPectrum:MARKer<x>:DELTa:X?`

Returns

`<NR3>`

Examples

`CALCULATE:LTE:CHSPECTRUM:MARKER1:DELTA:X?` might return 1.25000000E+6, indicating that the frequency of the specified delta marker is 1.25 MHz.

CALCulate:LTE:CHSPectrum:MARKer<x>:DELTa:Y? (Query Only)

Returns the amplitude for the specified delta marker on the trace in the LTE Channel Spectrum display. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement view: LTE Channel Spectrum

<x> = 1,2,3,4 is the marker.

Group

Calculate commands

Syntax

`CALCulate:LTE:CHSPectrum:MARKer<x>:DELTa:Y?`

Returns

`<NRf>`

Examples

`CALC:LTE:CHSPECTRUM:MARK1:DELT:Y?` might return 23.45, indicating that the amplitude of the specified delta marker is 23.45 dB.

CALCulate:LTE:CHSPectrum:MARKer<x>:MAXimum (No Query Form)

Positions the marker at the highest peak on the trace in the LTE Channel Spectrum display.

Conditions

Measurement view: LTE Channel Spectrum

<x> parameters: 0 to 4

Group

Calculate commands

Syntax

`CALCulate:LTE:CHSPectrum:MARKer<x>:MAXimum`

Arguments

None

Examples

`CALCULATE:LTE:CHSPECTRUM:MARKER1:MAXIMUM` moves marker 1 (M1) to the highest peak on the trace.

CALCulate:LTE:CHSPectrum:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the selected marker to the next peak to the left on the trace in the LTE Channel Spectrum display.

Conditions

Measurement view: LTE Channel Spectrum

<x> parameters: 0 to 4

Group

Calculate commands

Syntax

`CALCulate:LTE:CHSPectrum:MARKer<x>:PEAK:LEFT`

Arguments

None

Examples

`CALCULATE:LTE:CHSPECTRUM:MARKER1:PEAK:LEFT` moves marker 1 (M1) to the next peak to the left on the trace.

CALCulate:LTE:CHSPectrum:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the selected marker to the next peak to the right on the trace in the LTE Channel Spectrum display.

Conditions

Measurement view: LTE Channel Spectrum

<x> parameters: 0 to 4

Group

Calculate commands

Syntax

CALCulate:LTE:CHSPectrum:MARKer<x>:PEAK:RIGHt

Arguments

None

Examples

CALCULATE:LTE:CHSPECTRUM:MARKER1:PEAK:RIGHT moves marker 1 (M1) to the next peak to the right on the trace.

CALCulate:LTE:CHSPectrum:MARKer<x>:X

Sets or queries the horizontal position of the selected marker on the trace in the LTE Channel Spectrum display.

Conditions

Measurement view: LTE Channel Spectrum

<x> parameters: 0 to 4

Group

Calculate commands

Syntax

```
CALCulate:LTE:CHSPectrum:MARKer<x>:X <value>
CALCulate:LTE:CHSPectrum:MARKer<x>:X?
```

Arguments

<value>::=<NRf> specifies the horizontal position of the marker.

Returns

<NRf>

Examples

CALCULATE:LTE:CHSPECTRUM:MARKER1:X 800.000000000E+6 places marker 1 (M1) at 800 MHz on the trace.

CALCulate:LTE:CHSPectrum:MARKer<x>:Y? (Query Only)

Queries the amplitude of the selected marker on the trace in the LTE Channel Spectrum display.

Conditions

Measurement view: LTE Channel Spectrum

<x> parameters: 0 to 4

Group

Calculate commands

Syntax

CALCulate:LTE:CHSPectrum:MARKer<x>:Y?

Returns

<NRf>

Examples

CALCULATE:LTE:CHSPECTRUM:MARKER1:Y? might return –34.28, indicating that the amplitude of marker 1 (M1) is –34.28 dBm.

CALCulate:LTE:CONSte:MARKer<x>:FREQuency

Sets or returns the subcarrier number of the LTE symbol in the LTE Constellation display. This constellation shows only primary and secondary synchronization signals. Any input outside this range will show the marker set to -31 or +31.

Conditions

Measurement view: LTE Constellation

<x> parameters: 0 to 4

Range: –31 to +31

Group

Calculate commands

Syntax

```
CALCulate:LTE:CONSte:MARKer<x>:FREQuency <value>
CALCulate:LTE:CONSte:MARKer<x>:FREQuency?
```

Arguments

<value>::<NRf> specifies the subcarrier number of the specified marker

Returns

<NRf> The value type depends on where the marker is placed.

Examples

CALCULATE:LTE:CONSTE:MARKER2:FREQUENCY? might return +31, indicating that Marker 2 (M2) is positioned at the subcarrier +31 in the LTE Constellation display.

CALCulate:LTE:CONSte:MARKer<x>:MAGNitude? (Query Only)

Queries the marker magnitude of the selected marker in the LTE Constellation display.

Conditions

Measurement view: LTE Constellation

<x> parameters: 0 to 4

Group

Calculate commands

Syntax

```
CALCulate:LTE:CONSte:MARKer<x>:MAGNitude?
```

Returns

<NRf> The magnitude for the selected marker.

Examples

CALCULATE:LTE:CONSTE:MARKER2:MAGNITUDE might return "1.004" indicating readout of Marker 2 (M2) is "1.004".

CALCulate:LTE:CONSte:MARKer<x>:PHASe? (Query Only)

Queries the phase readout of the selected marker in the LTE Constellation display.

Conditions

Measurement view: LTE Constellation

Group

Calculate commands

Syntax

CALCulate:LTE:CONSte:MARKer<x>:PHASe?

Returns

<NRf> The phase readout for the selected marker.

Examples

CALCULATE:LTE:CONSTE:MARKER1:PHASE? might return 35.74, indicating the phase readout of Marker 1 (M1) is 35.74°

CALCulate:LTE:CONSte:MARKer<x>:TIME

Sets or queries the X position (Symbol Value) of the selected marker in the LTE Constellation display.

Conditions

Measurement view: LTE Constellation

<x> parameters: 0 to 4

For FDD, the range is 0 and 1.

For TDD the range is 0 and 3.

Group

Calculate commands

Syntax

CALCulate:LTE:CONSte:MARKer<x>:TIME <value>
CALCulate:LTE:CONSte:MARKer<x>:TIME?

Arguments

<value>::=<NRf> specifies the specified marker symbol number.

Returns

<NR1>

Examples

CALCULATE:LTE:CONSTE:MARKER<x>:TIME 0 places Marker 1 (M1) at the symbol 0 on range for time.

CALCulate:LTE:CONSte:MARKer<x>:TYPE? (Query Only)

Queries the data type of the selected marker in LTE Constellation display.The data type can either be “PSS” (Primary Synchronization Signal) or “SSS” (Secondary Synchronization Signal).

Conditions

Measurement view: LTE Constellation

<x> parameters: 0 to 4

Group

Calculate commands

Syntax

`CALCulate:LTE:CONSte:MARKer<x>:TYPE?`

Returns

`PSS` means that the data type for the selected marker is Primary Synchronization Signal

`SSS` means that the data type for the selected marker is Secondary Synchronization Signal

Examples

`CALCULATE:LTE:CONSTE:MARKER1:TYPE?` might return SSS, indicating that the data type for the Marker 1 (M1) is SSS.

CALCulate:LTE:PVTime:MARKer:<x>PEAK:HIGHer (No Query Form)

Moves the specified marker to the next peak higher in amplitude on the trace in the LTE Power vs Time display.

Conditions

Measurement view: LTE Power vs Time

Parameter <x>: 0 to 4

Group

Calculate commands

Syntax

`CALCulate:LTE:PVTime:MARKer:<x>PEAK:HIGHer`

Arguments

None

Examples

`CALCULATE:LTE:PVTIME:MARKER1:PEAK:HIGHER` moves marker 1 (M1) to the next peak higher in amplitude on the trace.

CALCulate:LTE:PVTime:MARKer<x>:DELTa:X? (Query Only)

Returns the frequency of the specified delta marker on the trace in the LTE Power vs Time display. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Conditions

Measurement view: LTE Power vs Time

<x> parameters: 1, 2, 3...(valid integral values range from 0 to the number of markers active for the display).

Querying an out-of-range value causes an execution error (-222, “Data out of range”).

Group

Calculate commands

Syntax

```
CALCulate:LTE:PVTime:MARKer<x>:DELTa:X?
```

Returns

```
<NR3>
```

Examples

`CALCULATE:LTE:PVTIME:MARKER1:DELTA:X?` might return `1.28000000E-6`, indicating that the frequency of the specified delta marker is 1.28 µs.

CALCulate:LTE:PVTime:MARKer<x>:DELTa:Y? (Query Only)

Returns the amplitude for the specified delta marker on the trace in the LTE Power vs Time display using the currently selected amplitude unit. You can set/query the amplitude unit with the SENSe:POWer:UNITs command. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Conditions

Measurement view: LTE Power vs Time

<x> parameters: 1, 2, 3...(valid integral values range from 0 to the number of markers active for the display).

Querying an out-of-range value causes an execution error (-222, "Data out of range").

Group

Calculate commands

Syntax

```
CALCulate:LTE:PVTime:MARKer<x>:DELTa:Y?
```

Returns

```
<NRf>
```

Examples

`CALCULATE:LTE:PVTIME:MARKER3:DELTA:Y?` might return –12.91, indicating that the amplitude of the delta marker 3 (M3) is –12.91 dB.

CALCulate:LTE:PVTime:MARKer<x>:MAXimum (No Query Form)

Positions the specified marker at the highest peak on the trace in the LTE Power vs Time display.

Conditions

Measurement view: LTE Power vs Time

<x> parameters: 0 to 4

Group

Calculate commands

Syntax

`CALCulate:LTE:PVTime:MARKer<x>:MAXimum`

Arguments

None

Examples

`CALCULATE:LTE:PVTIME:MARKER1:MAXIMUM` moves marker 1 (M1) to the highest peak on the trace.

CALCulate:LTE:PVTime:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the selected marker to the next peak to the left on the trace in the LTE Power vs Time display.

Conditions

Measurement view: LTE Power vs Time

<x> parameters: 0 to 4

Group

Calculate commands

Syntax

`CALCulate:LTE:PVTime:MARKer<x>:PEAK:LEFT`

Arguments

None

Examples

`CALCULATE:LTE:PVTIME:MARKER1:PEAK:LEFT` moves marker 1 (M1) to the next peak to the left on the trace.

CALCulate:LTE:PVTime:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the selected marker to the next peak lower in amplitude on the trace in the LTE Power vs Time display.

Conditions

Measurement view: LTE Power vs Time

<x> parameters: 0 to 4

Group

Calculate commands

Syntax

`CALCulate:LTE:PVTime:MARKer<x>:PEAK:LOWer`

Arguments

None

Examples

CALCULATE:LTE:PVTIME:MARKER1:PEAK:LOWER moves marker 1 (M1) to the next peak lower in amplitude on the trace.

CALCulate:LTE:PVTime:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the selected marker to the next peak to the right on the trace in the LTE Power vs Time display.

Conditions

Measurement view: LTE Power vs Time

<x> parameters: 0 to 4

Group

Calculate commands

Syntax

```
CALCulate:LTE:PVTime:MARKer<x>:PEAK:RIGHt
```

Arguments

None

Examples

CALCULATE:LTE:PVTIME:MARKER1:PEAK:RIGHT moves Marker 1 (M1) to the next peak to the right on the trace

CALCulate:LTE:PVTime:MARKer<x>:X

Sets or queries the horizontal position of the selected marker on the trace in the LTE Power vs Time display.

Conditions

Measurement view: LTE Power vs Time

<x> parameters: 0 to 4

Range: Start to Stop time (left to right edge of the horizontal axis). Using an out-of-range value causes an execution error (-222, "Data out of range").

Group

Calculate commands

Syntax

```
CALCulate:LTE:PVTime:MARKer<x>:X <NRf>
CALCulate:LTE:PVTime:MARKer<x>:X?
```

Arguments

<NRf> specifies the value of the horizontal position of the marker.

Returns

<NRf>

Examples

CALCULATE:LTE:PVTIME:MARKER3:X 1.000000000E-6 places Marker 3(M3) at 1 µs on the trace.

CALCulate:LTE:PVTime:MARKer<x>:Y? (Query Only)

Queries the amplitude of the selected marker on the trace in the LTE Power vs Time display. You can set/query the amplitude units with the `SENSe:POWer:UNITs` command.

Conditions

Measurement view: LTE Power vs Time

<x> parameters: 0 to 4 Querying an out-of-range value causes an execution error (-222, “Data out of range”).

Group

Calculate commands

Syntax

```
CALCulate:LTE:PVTime:MARKer<x>:Y?
```

Returns

`<NRf>` specifies the value of the horizontal position of the marker.

Examples

`CALCULATE:LTE:PVTIME:MARKER1:Y?` might return 33.393856E-3, indicating that the amplitude of marker 1 (M1) is 0.033 dBm.

CALCulate:MARKer:ADD (No Query Form)

Adds a marker. Every execution of this command adds a marker from MR, then M1 to M4, sequentially.

NOTE: If all markers are already turned on, the error message "Cannot add another marker" (execution error -200) is returned.

Conditions

Measurement views: All

Group

Calculate commands

Syntax

```
CALCulate:MARKer:ADD
```

Related Commands

CALCulate:MARKer:AOFF (on page 459)

CALCulate:MARKer:DELete (on page 459)

Arguments

None

Examples

`CALCULATE:MARKER:ADD` adds a marker.

CALCulate:MARKer:AOFF (No Query Form)

Turns off all markers.

Conditions

Measurement views: All

Group

Calculate commands

Syntax

`CALCulate:MARKer:AOFF`

Related Commands

CALCulate:MARKer:ADD (on page 458)

CALCulate:MARKer:DELete (on page 459)

Arguments

None

Examples

`CALCULATE:MARKER:AOFF` turns off all markers.

CALCulate:MARKer:DELete (No Query Form)

Deletes the last marker added.

NOTE: If all markers are turned off, the error message "Cannot delete another marker" (execution error -200) is returned.

Conditions

Measurement views: All

Group

Calculate commands

Syntax

`CALCulate:MARKer:DELete`

Related Commands

CALCulate:MARKer:ADD (on page 458)

CALCulate:MARKer:AOFF (on page 459)

Arguments

None

Examples

`CALCULATE:MARKER:DELETE` deletes the last marker added.

CALCulate:MARKer:DENSity:EXCursion

Sets or queries the minimum excursion of DPX signal density, or how far the density (hit count for pixels) must be above the surrounding noise to be detected as a peak. This command is effective for the DPX bitmap trace.

For marker peak up and marker peak down operations, the signal is considered to be a peak if it exceeds the excursion and the threshold set by the CALCulate:MARKer:DENSity:THReshold command.

For marker peak left and marker peak right operations, the signal is considered to be a peak if it first exceeds the amplitude excursion set by CALCulate:MARKer:PEAK:EXCursion, the amplitude threshold set by CALCulate:MARKer:PEAK:THReshold, the density excursion set by CALCulate:MARKer:DENSity:EXCursion and the density threshold set by CALCulate:MARKer:DENSity:THReshold.

Conditions

Measurement views: All

Group

Calculate commands

Syntax

```
CALCulate:MARKer:DENSity:EXCursion <number>
CALCulate:MARKer:DENSity:EXCursion?
```

Related Commands

CALCulate:MARKer:DENSity:THReshold (on page 461)

CALCulate:MARKer:PEAK:EXCursion (on page 463)

CALCulate:MARKer:PEAK:THReshold (on page 463)

Arguments

`<number>::=<NR1>` specifies the minimum excursion density.Range: 0 to 100%.

Examples

`CALCULATE:MARKER:DENSITY:EXCURSION 30` sets the minimum excursion density to 30.

CALCulate:MARKer:DENSity:SMOothing

Sets or queries the number of pixels squared for smoothing the DPX signal density. This command is effective for the DPX bitmap trace.

Conditions

Measurement views: All

Group

Calculate commands

Syntax

```
CALCulate:MARKer:DENSity:SMOothing <number>
CALCulate:MARKer:DENSity:SMOothing?
```

Arguments

`<number>::=<NR1>` specifies the number of pixels squared for smoothing the DPX signal density. Range: 1 to 20.

Examples

CALCULATE:MARKER:DENSITY:SMOOTHING5 sets the smoothing number to 5.

CALCulate:MARKer:DENSity:THReshold

Sets or queries the threshold of DPX signal density above which the density (hit count for pixels) is detected as a peak. This command is effective for the DPX bitmap trace.

Conditions

Measurement views: All

Group

Calculate commands

Syntax

```
CALCulate:MARKer:DENSity:THReshold <number>
CALCulate:MARKer:DENSity:THReshold?
```

Related Commands

CALCulate:MARKer:DENSity:EXCursion (on page 460)

Arguments

<number>::=<NR1> specifies the threshold density for detecting peaks.Range: 0 to 100%.

Examples

CALCULATE:MARKER:DENSITY:THRESHOLD 300 sets the threshold density to 300.

CALCulate:MARKer:DRAG:SEARch:STATe

Enables or disables peak searching when dragging a marker in a plot.

Conditions

Measurement views: All

Group

Calculate commands

Syntax

```
CALCulate:MARKer:DRAG:SEARch:STATe { OFF | ON | 0 | 1 }
CALCulate:MARKer:DRAG:SEARch:STATe?
```

Arguments

OFF or 0 disables peak searching in a plot.

ON or 1 enables peak searching in a plot.

Examples

CALCULATE:MARKER:DRAG:SEARCH:STATE OFF disables peak searching when dragging a marker in a plot.

CALCulate:MARKer:MODE

Sets or queries the marker mode.

Conditions

Measurement views: All

Group

Calculate commands

Syntax

```
CALCulate:MARKer:MODE { ABSolute | DELTa }
CALCulate:MARKer:MODE?
```

Arguments

`ABSolute` selects the absolute marker mode, in which the marker readout indicates the absolute value.

`DELTa` selects the delta marker mode, in which the marker readout indicates the relative value to the reference marker.

Examples

`CALCULATE:MARKER:MODEDELTa` selects the delta marker mode.

CALCulate:MARKer:MODE

Sets or queries the marker mode.

Conditions

Measurement views: All

Group

Calculate commands

Syntax

```
CALCulate:MARKer:MODE { ABSolute | DELTa | POWer }
CALCulate:MARKer:MODE?
```

Arguments

`ABSolute` selects the absolute marker mode, in which the marker readout indicates the absolute value.

`DELTa` selects the delta marker mode, in which the marker readout indicates the relative value to the selected marker.

`POWer` selects the power marker mode, in which the marker readout indicates the power value of the selected marker.

Examples

`CALCULATE:MARKER:MODE POWER` selects the power marker mode.

`CALCULATE:MARKER:MODE?` might return POW, meaning the Power is the assigned mode for the marker readout.

CALCulate:MARKer:NOISe:MODE

Sets or queries the view/hide status of the dBc/Hz Marker noise reading.

Group

Calculate commands

Syntax

```
CALCulate:MARKer:NOISe:MODE { 0 | 1 }
```

Arguments

0 disables the dBc/Hz reading.

1 enables the dBc/Hz reading.

Returns

0 means the dBc/Hz reading is not available.

1 means the dBc/Hz reading is not available.

Examples

CALCULATE:MARKER:NOISE:MODE 0 will disable the dBc/Hz marker noise reading.

CALCulate:MARKer:PEAK:EXCursion

Sets or queries the minimum excursion level (how far a signal must be above the surrounding noise to be detected as a peak). The signal is considered to be a peak if it exceeds the minimum excursion level and the threshold level set by the CALCulate:MARKer:PEAK:THReshold command.

Conditions

Measurement views: All

Group

Calculate commands

Syntax

```
CALCulate:MARKer:PEAK:EXCursion <value>
CALCulate:MARKer:PEAK:EXCursion?
```

Arguments

<value>::=<NRf> specifies the excursion level. Range: 0 to 100 dB.

Examples

CALCULATE:MARKER:PEAK:EXCURSION 10 sets the excursion level to 10 dB.

CALCulate:MARKer:PEAK:THReshold

Sets or queries the threshold level above which a signal is detected as a peak.

Conditions

Measurement views: All

Group

Calculate commands

Syntax

```
CALCulate:MARKer:PEAK:THReshold <value>
CALCulate:MARKer:PEAK:THReshold?
```

Arguments

<value>::=<NRf> specifies the threshold level for detecting peaks.Range: -170 to +130 dBm.

Examples

`CALCULATE:MARKER:PEAK:THRESHOLD -50` sets the threshold level to -50 dBm.

CALCulate:MCPower:MARKer<x>:DELTa:X? (Query Only)

Returns the delta marker frequency for the selected marker in the MCPR measurement.

The parameter <x> = 1 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Conditions

Measurement views: MCPR

Group

Calculate commands

Syntax

`CALCulate:MCPower:MARKer<x>:DELTa:X?`

Related Commands

CALCulate:MCPower:MARKer<x>:DELTa:Y? (on page 464)

Arguments

None

Returns

`<NRf>` Delta marker frequency for the selected marker.

Examples

`CALCULATE:MCPOWER:MARKER1:DELTA:X?` might return 1.28E+6, indicating that the delta marker frequency is 1.28 MHz.

CALCulate:MCPower:MARKer<x>:DELTa:Y? (Query Only)

Returns the delta marker amplitude for the selected marker in the MCPR measurement.

The parameter <x> = 1 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Conditions

Measurement views: MCPR

Group

Calculate commands

Syntax

`CALCulate:MCPower:MARKer<x>:DELTa:Y?`

Related Commands

CALCulate:MCPower:MARKer<x>:DELTa:X? (on page 464)

Arguments

None

Returns

<NRf> Delta marker amplitude for the selected marker.

Examples

CALCULATE:MCPOWER:MARKER1:DELTA:Y? might return 23.45, indicating that the delta marker amplitude is 23.45 dB.

CALCulate:MCPower:MARKer<x>:MAXimum (No Query Form)

Moves the selected marker to the highest peak on the trace in the MCPR measurement.

Conditions

Measurement views: MCPR

Group

Calculate commands

Syntax

CALCulate:MCPower:MARKer<x>:MAXimum

Arguments

None

Examples

CALCULATE:MCPOWER:MARKER1:MAXIMUM moves Marker 1 (M1) to the highest peak on the trace.

CALCulate:MCPower:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the selected marker to the next peak to the left on the trace in the MCPR measurement.

Conditions

Measurement views: MCPR

Group

Calculate commands

Syntax

CALCulate:MCPower:MARKer<x>:PEAK:LEFT

Related Commands

CALCulate:MCPower:MARKer<x>:PEAK:RIGHt (on page 466)

Arguments

None

Examples

CALCULATE:MCPOWER:MARKER1:PEAK:LEFT moves Marker 1 (M1) to the next peak to the left on the trace.

CALCulate:MCPower:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the selected marker to the next peak to the right on the trace in the MCPR measurement.

Conditions

Measurement views: MCPR

Group

Calculate commands

Syntax

```
CALCulate:MCPower:MARKer<x>:PEAK:RIGHt
```

Related Commands

CALCulate:MCPower:MARKer<x>:PEAK:LEFT (on page 465)

Arguments

None

Examples

`CALCULATE:MCPOWER:MARKER1:PEAK:RIGHT` moves Marker 1 (M1) to the next peak to the right on the trace.

CALCulate:MCPower:MARKer<x>:X

Sets or queries the horizontal position of the selected marker in the MCPR measurement.

Conditions

Measurement views: MCPR

Group

Calculate commands

Syntax

```
CALCulate:MCPower:MARKer<x>:X <value>
CALCulate:MCPower:MARKer<x>:X?
```

Related Commands

CALCulate:MCPower:MARKer<x>:Y? (on page 466)

Arguments

`<value>::=<NRf>` specifies the horizontal position of the marker.Range: Start to Stop frequency (left to right edge of the horizontal axis).Using an out-of-range value causes an execution error (-222, "Data out of range").

Examples

`CALCULATE:MCPOWER:MARKER1:X 800MHz` places Marker 1 (M1) at 800 MHz on the trace.

CALCulate:MCPower:MARKer<x>:Y? (Query Only)

Queries the marker amplitude of the selected marker in the MCPR measurement.

Conditions

Measurement views: MCPR

Group

Calculate commands

Syntax

`CALCulate:MCPower:MARKer<x>:Y?`

Related Commands

CALCulate:MCPower:MARKer<x>:X (on page 466)

Arguments

None

Returns

`<NRf>` Marker amplitude of the selected marker.

Examples

`CALCULATE:MCPOWER:MARKER1:Y?` might return -34.28, indicating Marker 1 (M1) is at -34.28 dBm.

CALCulate:MERRor[:MEASview<y>]:MARKer<x>:DELTa:X? (Query only)

Returns the delta marker time for the selected marker in the Magnitude error versus Time measurement.

The parameter <x> = 1 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Magnitude error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Magnitude error versus Time).

Group

Calculate commands

Syntax

`CALCulate:MERRor[:MEASview<y>]:MARKer<x>:DELTa:X?`

Related Commands

CALCulate:MERRor[:MEASview<y>]:MARKer<x>:DELTa:Y? (on page 468)

Arguments

None

Returns

`<NRf>` Delta marker amplitude for the selected marker in percent (%).

Examples

CALCulate:MERRor:MEASview2:MARKer1:DELTa:X? might return 3.84, indicating that the delta marker duration is 3.84.

CALCulate:MERRor[:MEASview<y>]:MARKer<x>:DELTa:Y? (Query only)

Returns the delta marker amplitude for the selected marker in the Magnitude error versus Time measurement.

The parameter <x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Magnitude error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Magnitude error versus Time).

Group

Calculate commands

Syntax

CALCulate:MERRor[:MEASview<y>]:MARKer<x>:DELTa:Y?

Related Commands

CALCulate:MERRor[:MEASview<y>]:MARKer<x>:DELTa:X? (on page 467)

Arguments

None

Returns

<NRf> Delta marker amplitude for the selected marker in percent (%).

Examples

CALCulate:MERRor:MEASview2:MARKer1:DELTa:Y? might return 3.84, indicating that the delta marker amplitude is 3.84%.

CALCulate:MERRor[:MEASview<y>]:MARKer<x>:MAXimum (No query form)

Moves the selected marker to the highest peak on the trace in the Magnitude error versus Time measurement.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Magnitude error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Magnitude error versus Time).

Group

Calculate commands

Syntax

```
CALCulate:MERRor[:MEASview<y>]:MARKer<x>:MAXimum
```

Arguments

None

Examples

`CALCulate:MERRor:MEASview2:MARKer1:MAXimum` moves Marker 1 (M1) to the highest peak on the trace.

CALCulate:MERRor[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (No query form)

Moves the selected marker to the next peak higher in amplitude on the Magnitude error versus Time trace.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Magnitude error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Magnitude error versus Time).

Group

Calculate commands

Syntax

```
CALCulate:MERRor[:MEASview<y>]:MARKer<x>:PEAK:HIGHer
```

Related Commands

CALCulate:MERRor[:MEASview<y>]:MARKer<x>:PEAK:LOWer (on page 470)

Arguments

None

Examples

`CALCUlate:MERRor:MEASview2:MARKer1:PEAK:HIGHER` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

CALCulate:MERRor[:MEASview<y>]:MARKer<x>:PEAK:LEFT (No query form)

Moves the selected marker to the next peak to the left on the trace in the Magnitude error versus Time measurement.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Magnitude error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Magnitude error versus Time).

Group

Calculate commands

Syntax

`CALCulate:MERRor[:MEASview<y>]:MARKer<x>:PEAK:LEFT`

Related Commands

CALCulate:MERRor[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (on page 471)

Arguments

None

Examples

`CALCulate:MERRor:MEASview2:MARKer1:PEAK:LEFT` moves Marker 1 (M1) to the next peak to the left on the trace.

CALCulate:MERRor[:MEASview<y>]:MARKer<x>:PEAK:LOWer (No query form)

Moves the selected marker to the next peak lower in amplitude on the Magnitude error versus Time trace.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Magnitude error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Magnitude error versus Time).

Group

Calculate commands

Syntax

`CALCulate:MERRor[:MEASview<y>]:MARKer<x>:PEAK:LOWer`

Related Commands

CALCulate:MERRor[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (on page 469)

Arguments

None

Examples

`CALCulate:MERRor:MEASview2:MARKer1:PEAK:LOWer` moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

CALCulate:MERRor[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (No query form)

Moves the selected marker to the next peak to the right on the trace in the Magnitude error versus Time measurement.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Magnitude error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Magnitude error versus Time).

Group

Calculate commands

Syntax

```
CALCulate:MERRor[:MEASview<y>]:MARKer<x>:PEAK:RIGHt
```

Related Commands

CALCulate:MERRor[:MEASview<y>]:MARKer<x>:PEAK:LEFT (on page 469)

Arguments

None

Examples

`CALCulate:MERRor:MEASview2:MARKer1:PEAK:RIGHT` moves Marker 1 (M1) to the next peak to the right on the trace.

CALCulate:MERRor[:MEASview<y>]:MARKer<x>:X

Sets or queries the horizontal position of the selected marker in the Magnitude error versus Time measurement.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Magnitude error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Magnitude error versus Time).

Group

Calculate commands

Syntax

```
CALCulate:MERRor[:MEASview<y>]:MARKer<x>:X <value>
CALCulate:MERRor[:MEASview<y>]:MARKer<x>:X?
```

Arguments

<value>::=<NRf> specifies the horizontal position of the marker. Range: (analysis offset) to [(analysis offset) + (analysis length)].

The unit can be changed by the [SENSe]:DDEMod:TIMe:UNITs command.

Examples

CALCUlate:MERRor:MEASview2:MARKer1:X 1.5µs places Marker 1 (M1) at 1.5 µs on the trace.

CALCulate:MERRor[:MEASview<y>]:MARKer<x>:Y? (Query only)

Queries the marker amplitude of the selected marker in the Magnitude error versus Time measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Magnitude error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Magnitude error versus Time).

Group

Calculate commands

Syntax

CALCulate:MERRor[:MEASview<y>]:MARKer<x>:Y?

Related Commands

CALCulate:MERRor[:MEASview<y>]:MARKer<x>:X (on page 471)

Arguments

None

Returns

<NRf> Marker amplitude of the selected marker in percent (%).

Examples

CALCulate:MERRor:MEASview2:MARKer1:Y? might return 5.34, indicating Marker 1 (M1) is at 5.34%.

CALCulate:NOISe:FIGure:MARKer<x>:DELTa:X? (Query Only)

Queries the frequency difference between MR (reference marker) and the selected marker.

Conditions

Measurement view: Noise Figure

The parameter <x> = 1 to 4.

Group

Calculate commands

Syntax

CALCulate:NOISe:FIGure:MARKer<x>:DELTa:X?

Related Commands

CALCulate:NOISe:FIGure:MARKer<x>:Y? (on page 477)

Returns

<NRf> is the frequency difference value between MR and the selected marker.

Examples

CALCULATE:NOISE:FIGURE:MARKER1:DELTA:X? might return 0.0000,indicating the frequency difference between MR (reference marker) and the selected marker (M1) is 0.0000 dB.

CALCulate:NOISe:FIGure:MARKer<x>:DELTa:Y? (Query only)

Queries the power difference value between MR (reference marker) and the selected marker.

Conditions

Measurement view: Noise Figure

The parameter <x> = 1 to 4.

Group

Calculate commands

Syntax

CALCulate:NOISe:FIGure:MARKer<x>:DELTa:Y?

Related Commands

CALCulate:NOISe:FIGure:MARKer<x>:X? (on page 477)

Returns

<NRf> is the power difference value between MR and the selected marker.

Examples

CALCULATE:NOISE:FIGURE:MARKER1:DELTA:Y? might return18.7647705078, indicating the difference between MR (reference marker) and the selected marker (M1) is 18.76 dB.

CALCulate:NOISe:FIGure:MARKer<x>:MAXimum (No Query Form)

Places the selected marker at the maximum value on the peak of the trace.

Conditions

Measurement view: Noise Figure

The parameter <x> = 0 to 4

Group

Calculate commands

Syntax

CALCulate:NOISe:FIGure:MARKer<x>:MAXimum

Related Commands

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:HIGHer (on page 474)

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:LOWer (on page 475)

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:LEFT (on page 474)

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:RIGHt (on page 475)

Arguments

None

Returns

None

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:HIGHer (No Query Form)

Places the selected marker at the next highest peak on the trace.

Conditions

Measurement view: Noise Figure

The parameter <x> = 0 to 4

Group

Calculate commands

Syntax

```
CALCulate:NOISe:FIGure:MARKer<x>:PEAK:HIGHer
```

Related Commands

CALCulate:NOISe:GAIN:MARKer<x>:MAXimum? (on page 479)

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:LOWer (on page 475)

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:LEFT (on page 474)

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:RIGHt (on page 475)

Arguments

None

Returns

None

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:LEFT (No Query Form)

Places the selected marker at the next peak on the trace to the left of the selected marker.

Conditions

Measurement view: Noise Figure

The parameter <x> = 0 to 4

Group

Calculate commands

Syntax

```
CALCulate:NOISe:FIGure:MARKer<x>:PEAK:LEFT
```

Related Commands

CALCulate:NOISe:FIGure:MARKer<x>:MAXimum (on page 473)

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:HIGHer (on page 474)

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:LOWer (on page 475)

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:RIGHt (on page 475)

Arguments

None

Returns

None

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:LOWer (No Query Form)

Places the selected marker at the next lowest peak on the trace.

Conditions

Measurement view: Noise Figure

The parameter <x> = 0 to 4

Group

Calculate commands

Syntax

```
CALCulate:NOISe:FIGure:MARKer<x>:PEAK:LOWer
```

Related Commands

CALCulate:NOISe:FIGure:MARKer<x>:MAXimum (on page 473)

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:HIGHer (on page 474)

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:LEFT (on page 474)

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:RIGHt (on page 475)

Arguments

None

Returns

None

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:RIGHt (No Query Form)

This command places the specified marker on a specified trace. The query form returns the trace on which the marker resides.

Conditions

Measurement view: Noise Figure

The parameter <x> = 0 to 4

Group

Calculate commands

Syntax

```
CALCulate:NOISe:FIGure:MARKer<x>:PEAK:RIGHt
```

Related Commands

CALCulate:NOISe:FIGure:MARKer<x>:MAXimum (on page 473)

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:HIGHer (on page 474)

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:LOWer (on page 475)

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:LEFT (on page 474)

Arguments

None

Returns

None

CALCulate:NOISe:FIGure:MARKer<x>:TRACe

Sets or queries which trace the specified marker is attached to.

Conditions

Measurement view: Noise Figure

The parameter <x> = 0 to 4

Group

Calculate commands

Syntax

```
CALCulate:NOISe:FIGure:MARKer<x>:TRACe {TRACE1|TRACE2|TRACE3}
CALCulate:NOISe:FIGure:MARKer<x>:TRACe?
```

Related Commands

CALCulate:NOISe:FIGure:MARKer<x>:MAXimum (on page 473)

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:HIGHer (on page 474)

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:LOWer (on page 475)

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:LEFT (on page 474)

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:RIGHt (on page 475)

Arguments

`TRACE1` moves the specified marker to trace 1.

`TRACE2` moves the specified marker to trace 2.

TRACE3 moves the specified marker to trace 3.

Returns

TRACE1 means the specified marker is on trace 1.

TRACE2 means the specified marker is on trace 2.

TRACE3 means the specified marker is on trace 3.

Examples

CALCULATE:NOISE:FIGURE:MARKER1:TRACE? might return TRACE2, indicating that M1 (marker 1) is on trace 2.

CALCulate:NOISe:FIGure:MARKer<x>:X? (Query Only)

Queries the frequency of the specified marker.

Conditions

Measurement view: Noise Figure

The parameter <x> = 0 to 4

Group

Calculate commands

Syntax

CALCulate:NOISe:FIGure:MARKer<x>:X?

Related Commands

CALCulate:NOISe:FIGure:MARKer<x>:Y? (on page 477)

Returns

<NR3> = a floating point value with an exponent. This is the frequency of the specified marker

Examples

CALCULATE:NOISE:FIGURE:MARKER1:X? might return 608.0000000000E+6,indicating the frequency of M1 (marker 1) is 608 MHz.

CALCulate:NOISe:FIGure:MARKer<x>:Y? (Query Only)

Queries the vertical position of the specified marker.

Conditions

Measurement view: Noise Figure

The parameter <x> = 0 to 4

Group

Calculate commands

Syntax

CALCulate:NOISe:FIGure:MARKer<x>:Y?

Related Commands

CALCulate:NOISe:FIGure:MARKer<x>:X? (on page 477)

Returns

<NR2> = a floating point value without an exponent. This is the vertical position(dB) of the specified marker.

Examples

CALCULATE:NOISE:FIGURE:MARKER1:Y? might return 18.7647705078,indicating the value of M1 (marker 1) is 18.76 dB.

CALCulate:NOISe:GAIN:MARKer<x>:DELTa:X? (Query Only)

Query returns the frequency difference between MR (reference marker) and the selected marker.

Conditions

Measurement view: Gain

The parameter <x> = 1 to 4.

Group

Calculate commands

Syntax

CALCulate:NOISe:GAIN:MARKer<x>:DELTa:X?

Related Commands

CALCulate:NOISe:GAIN:MARKer<x>:DELTa:Y? (on page 478)

Returns

<NR2> = a floating point value without an exponent. This is the frequency difference value between MR and the selected marker.

Examples

CALCULATE:NOISE:GAIN:MARKER1:DELTA:X? might return 100.0000, indicating the frequency difference between MR (reference marker) and the selected marker (M1) is 100 dB.

CALCulate:NOISe:GAIN:MARKer<x>:DELTa:Y? (Query Only)

Query returns the power difference value between MR (reference marker) and the selected marker.

Conditions

Measurement view: Gain

The parameter <x> = 1 to 4.

Group

Calculate commands

Syntax

CALCulate:NOISe:GAIN:MARKer<x>:DELTa:Y?

Related Commands

CALCulate:NOISe:GAIN:MARKer<x>:DELTa:X? (on page 478)

Returns

<NR2> = a floating point value without an exponent. This is the power difference value between MR and the selected marker.

Examples

CALCULATE:NOISE:GAIN:MARKER1:DELTA:Y? might return 18.7647705078, indicating the difference between MR (reference marker) and the selected marker(M1) is 18.76 dB.

CALCulate:NOISe:GAIN:MARKer<x>:MAXimum? (Query Only)

Places the selected marker at the maximum value on the peak of the trace.

Conditions

Measurement view: Gain

The parameter <x> = 0 to 4

Group

Calculate commands

Syntax

CALCulate:NOISe:GAIN:MARKer<x>:MAXimum?

Related Commands

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:HIGHer? (on page 479)

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:LEFT? (on page 480)

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:RIGHt? (on page 481)

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:LOWer? (on page 480)

Arguments

None

Returns

None

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:HIGHer? (Query Only)

Places the selected marker at the next highest peak on the trace.

Conditions

Measurement view: Gain

The parameter <x> = 0 to 4

Group

Calculate commands

Syntax

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:HIGHer?

Related Commands

CALCulate:NOISe:GAIN:MARKer<x>:MAXimum? (on page 479)

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:LEFT? (on page 480)

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:RIGHt? (on page 481)

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:LOWer? (on page 480)

Arguments

None

Returns

None

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:LEFT? (Query Only)

Places the selected marker at the next peak on the trace to the left of the selected marker.

Conditions

Measurement view: Gain

The parameter <x> = 0 to 4

Group

Calculate commands

Syntax

```
CALCulate:NOISe:GAIN:MARKer<x>:PEAK:LEFT?
```

Related Commands

CALCulate:NOISe:GAIN:MARKer<x>:MAXimum? (on page 479)

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:HIGHer? (on page 479)

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:RIGHt? (on page 481)

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:LOWer? (on page 480)

Arguments

None

Returns

None

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:LOWer? (Query Only)

Places the selected marker at the next lowest peak on the trace.

Conditions

Measurement view: Gain

The parameter <x> = 0 to 4

Group

Calculate commands

Syntax

```
CALCulate:NOISe:GAIN:MARKer<x>:PEAK:LOWer?
```

Related Commands

CALCulate:NOISe:GAIN:MARKer<x>:MAXimum? (on page 479)

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:HIGHer? (on page 479)

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:LEFT? (on page 480)

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:RIGHt? (on page 481)

Arguments

None

Returns

None

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:RIGHt? (Query Only)

Places the specified marker on a specified trace. The query form returns the trace on which the marker resides.

Conditions

Measurement view: Gain

The parameter <x> = 0 to 4

Group

Calculate commands

Syntax

```
CALCulate:NOISe:GAIN:MARKer<x>:PEAK:RIGHt?
```

Related Commands

CALCulate:NOISe:GAIN:MARKer<x>:MAXimum? (on page 479)

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:HIGHer? (on page 479)

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:LEFT? (on page 480)

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:LOWer? (on page 480)

Arguments

None

Returns

None

CALCulate:NOISe:GAIN:MARKer<x>:TRACe

Sets or queries which trace the specified marker is attached to.

Conditions

Measurement view: Gain

The parameter <x> = 0 to 4

Group

Calculate commands

Syntax

```
CALCulate:NOISe:GAIN:MARKer<x>:TRACe { TRACE1 | TRACE2 | TRACE3}
CALCulate:NOISe:GAIN:MARKer<x>:TRACe?
```

Arguments

`TRACE1` moves the specified marker to trace 1.

`TRACE2` moves the specified marker to trace 2.

`TRACE3` moves the specified marker to trace 3.

Returns

TRACE1 means the specified marker is on trace 1.

TRACE2 means the specified marker is on trace 2.

TRACE3 means the specified marker is on trace 3.

Examples

`CALCULATE:NOISE:GAIN:MARKER1:TRACE?` might return TRACE2, indicating that M1 (marker 1) is on trace 2.

CALCulate:NOISe:GAIN:MARKer<x>:X? (Query Only)

This command queries the frequency of the specified marker.

Conditions

Measurement view: Gain

The parameter <x> = 0 to 4

Group

Calculate commands

Syntax

```
CALCulate:NOISe:GAIN:MARKer<x>:X?
```

Related Commands

CALCulate:NOISe:GAIN:MARKer<x>:Y? (on page 483)

Returns

`<NR3>`= a floating point value with an exponent. This is the frequency of the specified marker.

Examples

`CALCULATE:NOISE:GAIN:MARKER1:X?` might return `608.0000000000E+6`, indicating the frequency of M1 (marker 1) is 608 MHz.

CALCulate:NOISe:GAIN:MARKer<x>:Y? (Query Only)

Queries the vertical position (dB) of the specified marker.

Conditions

Measurement view: Gain

The parameter <x> = 0 to 4

Group

Calculate commands

Syntax

`CALCulate:NOISe:GAIN:MARKer<x>:Y?`

Related Commands

CALCulate:NOISe:TEMPerature:MARKer<x>:DELTa:X? (on page 483)

Returns

`<NR2>` = a floating point value without an exponent. This is the vertical position(dB) of the specified marker.

Examples

`CALCULATE:NOISE:GAIN:MARKER1:Y?` might return `18.7647705078`, indicating the value of M1 (marker 1) is 18.76 dB.

CALCulate:NOISe:TEMPerature:MARKer<x>:DELTa:X? (Query Only)

eturns the frequency difference between MR (reference marker) and the selected marker.

Conditions

Measurement view: Noise Temperature

The parameter <x> = 1 to 4.

Group

Calculate commands

Syntax

`CALCulate:NOISe:TEMPerature:MARKer<x>:DELTa:X?`

Related Commands

CALCulate:NOISe:TEMPerature:MARKer<x>:DELTa:Y? (on page 484)

Returns

<NR2> = a floating point value without an exponent. This is the frequency difference value between MR and the selected marker.

Examples

`CALC:NOIS:TEMP:MARK1:DELT:X?` might return 100.0000, indicating the frequency difference between MR (reference marker) and the selected marker(M1) is 100 dB.

CALCulate:NOISe:TEMPerature:MARKer<x>:DELTa:Y? (Query Only)

Returns the temperature difference between MR (reference marker) and the selected marker.

Conditions

Measurement view: Noise Temperature

The parameter <x> = 1 to 4.

Group

Calculate commands

Syntax

`CALCulate:NOISe:TEMPerature:MARKer<x>:DELTa:Y?`

Related Commands

CALCulate:NOISe:TEMPerature:MARKer<x>:DELTa:X? (on page 483)

Returns

`<NR3>` = a floating point value with an exponent. This is the temperature difference value between MR and the selected marker.

Examples

`CALC:NOIS:TEMP:MARK1:DELT:Y?` might return 2.3663560181E+3,indicating the difference between MR (reference marker) and the selected marker(M1) is 2.36 kK.

CALCulate:NOISe:TEMPerature:MARKer<x>:MAXimum? (Query Only)

Places the selected marker at the maximum value on the peak of the trace.

Conditions

Measurement view: Noise Temperature

The parameter <x> = 0 to 4

Group

Calculate commands

Syntax

`CALCulate:NOISe:TEMPerature:MARKer<x>:MAXimum?`

Related Commands

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:HIGHer? (on page 485)

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:LEFT? (on page 485)

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:LOWer? (on page 486)

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:RIGHt? (on page 486)

Arguments

None

Returns

None

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:HIGHer? (Query Only)

Places the selected marker at the highest peak on the trace.

Conditions

Measurement view: Noise Temperature

The parameter <x> = 0 to 4

Group

Calculate commands

Syntax

```
CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:HIGHer?
```

Related Commands

CALCulate:NOISe:TEMPerature:MARKer<x>:MAXimum? (on page 484)

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:LEFT? (on page 485)

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:LOWer? (on page 486)

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:RIGHt? (on page 486)

Arguments

None

Returns

None

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:LEFT? (Query Only)

Places the selected marker at the next peak on the trace to the left of the selected marker.

Conditions

Measurement view: Noise Temperature

The parameter <x> = 0 to 4

Group

Calculate commands

Syntax

```
CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:LEFT?
```

Related Commands

CALCulate:NOISe:TEMPerature:MARKer<x>:MAXimum? (on page 484)

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:HIGHer? (on page 485)

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:LOWer? (on page 486)

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:RIGHt? (on page 486)

Arguments

None

Returns

None

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:LOWer? (Query Only)

Places the selected marker at the next lowest peak on the trace.

Conditions

Measurement view: Noise Temperature

The parameter <x> = 0 to 4

Group

Calculate commands

Syntax

```
CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:LOWer?
```

Related Commands

CALCulate:NOISe:TEMPerature:MARKer<x>:MAXimum? (on page 484)

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:HIGHer? (on page 485)

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:LEFT? (on page 485)

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:RIGHt? (on page 486)

Arguments

None

Returns

None

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:RIGHt? (Query Only)

This command places the specified marker on a specified trace. The query form returns the trace on which the marker resides.

Conditions

Measurement view: Noise Temperature

The parameter <x> = 0 to 4

Group

Calculate commands

Syntax

```
CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:RIGHt?
```

Related Commands

CALCulate:NOISe:TEMPerature:MARKer<x>:MAXimum? (on page 484)

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:HIGHer? (on page 485)

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:LEFT? (on page 485)

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:LOWer? (on page 486)

Arguments

None

Returns

None

CALCulate:NOISe:TEMPerature:MARKer<x>:TRACe

Sets or queries which trace the specified marker is attached to.

Conditions

Measurement view: Noise Temperature

The parameter <x> = 0 to 4

Group

Calculate commands

Syntax

```
CALCulate:NOISe:TEMPerature:MARKer<x>:TRACe { TRACE1 | TRACE2| TRACE3 }
CALCulate:NOISe:TEMPerature:MARKer<x>:TRACe?
```

Arguments

`TRACE1` moves the specified marker to trace 1.

`TRACE2` moves the specified marker to trace 2.

`TRACE3` moves the specified marker to trace 3.

Returns

TRACE1 means the specified marker is on trace 1.

TRACE2 means the specified marker is on trace 2.

TRACE3 means the specified marker is on trace 3.

Examples

`CALC:NOIS:TEMP:MARK1:TRAC?` might return TRACE2, indicating that M1(marker 1) is on trace 2.

CALCulate:NOISe:TEMPerature:MARKer<x>:X? (Query Only)

Queries the frequency of the specified marker.

Conditions

Measurement view: Noise Temperature

The parameter <x> = 0 to 4

Group

Calculate commands

Syntax

`CALCulate:NOISe:TEMPerature:MARKer<x>:X?`

Related Commands

CALCulate:NOISe:TEMPerature:MARKer<x>:Y? (on page 488)

Returns

`<NR2>` = a floating point value without an exponent. This is the frequency of the specified marker.

Examples

`CALC:NOIS:TEMP:MARK1:X?` might return `608.0000000000E+6`, indicating the frequency of M1 (marker 1) is 608 MHz.

CALCulate:NOISe:TEMPerature:MARKer<x>:Y? (Query Only)

Queries the temperature (Kelvin) at the specified marker.

Conditions

Measurement view: Noise Temperature

The parameter <x> = 0 to 4

Group

Calculate commands

Syntax

`CALCulate:NOISe:TEMPerature:MARKer<x>:Y?`

Related Commands

CALCulate:NOISe:TEMPerature:MARKer<x>:X? (on page 488)

Returns

`<NR3>` = a floating point value with an exponent. This is the temperature at the specified marker.

Examples

`CALC:NOIS:TEMP:MARK1:Y?` might return `1.4544410400E+3`, indicating the value at M1 (marker 1) is 1.45 kK.

CALCulate:NOISe:YFACtor:MARKer<x>:DELTa:X? (Query Only)

Queries the frequency difference between MR (reference marker) and the selected marker.

Conditions

Measurement view: Y Factor

The parameter <x> = 1 to 4.

Group

Calculate commands

Syntax

`CALCulate:NOISe:YFACtor:MARKer<x>:DELTa:X?`

Related Commands

CALCulate:NOISe:YFACtor:MARKer<x>:DELTa:Y? (on page 489)

Returns

`<NR2>` = a floating point value without an exponent. This is the frequency difference value between MR and the selected marker.

Examples

`CALC:NOIS:YFAC:MARK1:DELT:X?` might return 100.0000, indicating the frequency difference between MR (reference marker) and the selected marker(M1) is 100 dB.

CALCulate:NOISe:YFACtor:MARKer<x>:DELTa:Y? (Query Only)

Queries the power difference value between MR (reference marker) and the selected marker.

Conditions

Measurement view: Y Factor

The parameter <x> = 1 to 4.

Group

Calculate commands

Syntax

`CALCulate:NOISe:YFACtor:MARKer<x>:DELTa:Y?`

Related Commands

CALCulate:NOISe:YFACtor:MARKer<x>:DELTa:X? (on page 489)

Returns

`<NR2>` = a floating point value without an exponent. This is the power difference value between MR and the selected marker.

Examples

`CALC:NOIS:YFAC:MARK1:DELT:Y?` might return 18.7647705078, indicating the difference between MR (reference marker) and the selected marker (M1) is 18.76 dB.

CALCulate:NOISe:YFACtor:MARKer<x>:MAXimum (No Query Form)

Places the selected marker at the maximum value on the peak of the trace.

Conditions

Measurement view: Y Factor

The parameter <x> = 0 to 4

Group

Calculate commands

Syntax

```
CALCulate:NOISe:YFACtor:MARKer<x>:MAXimum
```

Related Commands

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:HIGHer? (on page 490)

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:LEFT? (on page 491)

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:LOWer? (on page 491)

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:RIGHt? (on page 492)

Arguments

None

Returns

None

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:HIGHer? (Query Only)

This command places the selected marker at the next highest peak on the trace.

Conditions

Measurement view: Y Factor

The parameter <x> = 0 to 4

Group

Calculate commands

Syntax

```
CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:HIGHer?
```

Related Commands

CALCulate:NOISe:YFACtor:MARKer<x>:MAXimum (on page 490)

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:LEFT? (on page 491)

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:LOWer? (on page 491)

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:RIGHt? (on page 492)

Arguments

None

Returns

None

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:LEFT? (Query Only)

This command places the selected marker at the next peak on the trace to the left of the selected marker.

Conditions

Measurement view: Y Factor

The parameter <x> = 0 to 4

Group

Calculate commands

Syntax

```
CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:LEFT?
```

Related Commands

CALCulate:NOISe:YFACtor:MARKer<x>:MAXimum (on page 490)

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:HIGHer? (on page 490)

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:LOWer? (on page 491)

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:RIGHt? (on page 492)

Arguments

None

Returns

None

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:LOWer? (Query Only)

Places the selected marker at the next lowest peak on the trace.

Conditions

Measurement view: Y Factor

The parameter <x> = 0 to 4

Group

Calculate commands

Syntax

```
CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:LOWer?
```

Related Commands

CALCulate:NOISe:YFACtor:MARKer<x>:MAXimum (on page 490)

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:HIGHer? (on page 490)

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:LEFT? (on page 491)

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:RIGHt? (on page 492)

Arguments

None

Returns

None

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:RIGHt? (Query Only)

Places the specified marker on a specified trace. The query form returns the trace on which the marker resides.

Conditions

Measurement view: Y Factor

The parameter <x> = 0 to 4

Group

Calculate commands

Syntax

```
CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:RIGHt?
```

Related Commands

CALCulate:NOISe:YFACtor:MARKer<x>:MAXimum (on page 490)

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:HIGHer? (on page 490)

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:LEFT? (on page 491)

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:LOWer? (on page 491)

Arguments

None

Returns

None

CALCulate:NOISe:YFACtor:MARKer<x>:TRACe

Sets or queries which trace the specified marker is attached to.

Conditions

Measurement view: Y Factor

The parameter <x> = 0 to 4

Group

Calculate commands

Syntax

```
CALCulate:NOISe:YFACtor:MARKer<x>:TRACe {TRACE1|TRACE2|TRACE3}
CALCulate:NOISe:YFACtor:MARKer<x>:TRACe?
```

Arguments

TRACE1 moves the specified marker to trace 1.

TRACE2 moves the specified marker to trace 2.

TRACE3 moves the specified marker to trace 3.

Returns

TRACE1 means the specified marker is on trace 1.

TRACE2 means the specified marker is on trace 2.

TRACE3 means the specified marker is on trace 3.

Examples

CALC:NOIS:YFAC:MARK1:TRAC? might return TRACE2, indicating that M1(marker 1) is on trace 2.

CALCulate:NOISe:YFACtor:MARKer<x>:X? (Query Only)

Queries the frequency of the specified marker.

Conditions

Measurement view: Y Factor

The parameter <x> = 0 to 4

Group

Calculate commands

Syntax

CALCulate:NOISe:YFACtor:MARKer<x>:X?

Related Commands

CALCulate:NOISe:YFACtor:MARKer<x>:Y? (on page 493)

Returns

<NR3> = a floating point value with an exponent. This is the frequency of the specified marker.

Examples

CALC:NOIS:YFAC:MARK1:X? might return 608.0000000000E+6, indicating the frequency of M1 (marker 1) is 608 MHz.

CALCulate:NOISe:YFACtor:MARKer<x>:Y? (Query Only)

This command queries the vertical position (dB) of the specified marker.

Conditions

Measurement view: Y Factor

The parameter <x> = 0 to 4

Group

Calculate commands

Syntax

CALCulate:NOISe:YFACtor:MARKer<x>:Y?

Related Commands

CALCulate:NOISe:YFACtor:MARKer<x>:X? (on page 493)

Returns

<NR2> = a floating point value without an exponent. This is the dB of the specified marker.

Examples

CALCULATE:NOISE:YFACTOR:MARKER1:Y? might return 18.7647705078,indicating the value of M1 (marker 1) is 18.76 dB.

CALCulate:OBWidth:MARKer<x>:DELTa:X? (Query Only)

Returns the delta marker frequency for the selected marker in the Occupied Bandwidth measurement.

The parameter <x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: Occupied Bandwidth

Group

Calculate commands

Syntax

CALCulate:OBWidth:MARKer<x>:DELTa:X?

Related Commands

CALCulate:OBWidth:MARKer<x>:DELTa:Y? (on page 494)

Arguments

None

Returns

<NRf> Delta marker frequency for the selected marker.

Examples

CALCULATE:OBWIDTH:MARKER1:DELTA:X? might return 1.28E+6, indicating that the delta marker frequency is 1.28 MHz.

CALCulate:OBWidth:MARKer<x>:DELTa:Y? (Query Only)

Returns the delta marker amplitude for the selected marker in the Occupied Bandwidth measurement.

The parameter <x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: Occupied Bandwidth

Group

Calculate commands

Syntax

```
CALCulate:OBWidth:MARKer<x>:DELTa:Y?
```

Related Commands

CALCulate:OBWidth:MARKer<x>:DELTa:X? (on page 494)

Arguments

None

Returns

<NRf> Delta marker amplitude for the selected marker.

Examples

CALCULATE:OBWIDTH:MARKER1:DELTA:Y? might return 23.45, indicating that the delta marker amplitude is 23.45 dB.

CALCulate:OBWidth:MARKer<x>:MAXimum (No Query Form)

Moves the selected marker to the highest peak on the trace in the Occupied Bandwidth measurement.

Conditions

Measurement views: Occupied Bandwidth

Group

Calculate commands

Syntax

```
CALCulate:OBWidth:MARKer<x>:MAXimum
```

Arguments

None

Examples

CALCULATE:OBWIDTH:MARKER1:MAXIMUM moves Marker 1 (M1) to the highest peak on the trace.

CALCulate:OBWidth:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the selected marker to the next peak higher in amplitude on the Occupied Bandwidth trace.

Conditions

Measurement views: Occupied Bandwidth

Group

Calculate commands

Syntax

```
CALCulate:OBWidth:MARKer<x>:PEAK:HIGHer
```

Related Commands

CALCulate:OBWidth:MARKer<x>:PEAK:LOWer (on page 496)

Arguments

None

Examples

`CALCULATE:OBWIDTH:MARKER1:PEAK:HIGHER` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

CALCulate:OBWidth:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the selected marker to the next peak to the left on the trace in the Occupied Bandwidth measurement.

Conditions

Measurement views: Occupied Bandwidth

Group

Calculate commands

Syntax

`CALCulate:OBWidth:MARKer<x>:PEAK:LEFT`

Related Commands

CALCulate:OBWidth:MARKer<x>:PEAK:RIGHt (on page 497)

Arguments

None

Examples

`CALCULATE:OBWIDTH:MARKER1:PEAK:LEFT` moves Marker 1 (M1) to the next peak to the left on the trace.

CALCulate:OBWidth:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the selected marker to the next peak lower in amplitude on the Occupied Bandwidth trace.

Conditions

Measurement views: Occupied Bandwidth

Group

Calculate commands

Syntax

`CALCulate:OBWidth:MARKer<x>:PEAK:LOWer`

Related Commands

CALCulate:OBWidth:MARKer<x>:PEAK:HIGHer (on page 495)

Arguments

None

Examples

CALCULATE:OBWIDTH:MARKER1:PEAK:LOWER moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

CALCulate:OBWidth:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the selected marker to the next peak to the right on the trace in the Occupied Bandwidth measurement.

Conditions

Measurement views: Occupied Bandwidth

Group

Calculate commands

Syntax

CALCulate:OBWidth:MARKer<x>:PEAK:RIGHt

Related Commands

CALCulate:OBWidth:MARKer<x>:PEAK:LEFT (on page 496)

Arguments

None

Examples

CALCULATE:OBWIDTH:MARKER1:PEAK:RIGHT moves Marker 1 (M1) to the next peak to the right on the trace.

CALCulate:OBWidth:MARKer<x>[:SET]:CENTer (No Query Form)

Sets the center frequency to the value at the marker position in the Occupied Bandwidth measurement.

Conditions

Measurement views: Occupied Bandwidth

Group

Calculate commands

Syntax

CALCulate:OBWidth:MARKer<x>[:SET]:CENTer (on page 497)

Arguments

None

Examples

CALCULATE:OBWIDTH:MARKER1:SET:CENTER sets the center frequency to the value at the Marker 1 position.

CALCulate:OBWidth:MARKer<x>:X

Sets or queries the horizontal position of the selected marker in the Occupied Bandwidth measurement.

Conditions

Measurement views: Occupied Bandwidth

Group

Calculate commands

Syntax

```
CALCulate:OBWidth:MARKer<x>:X <value>
CALCulate:OBWidth:MARKer<x>:X?
```

Related Commands

CALCulate:OBWidth:MARKer<x>:Y? (on page 498)

Arguments

`<value>::=<NRf>` specifies the horizontal position of the marker.Range: Start to Stop frequency (left to right edge of the horizontal axis).Using an out-of-range value causes an execution error (-222, "Data out of range").

Examples

`CALCULATE:OBWIDTH:MARKER1:X 800MHz` places Marker 1 (M1) at 800 MHz on the trace.

CALCulate:OBWidth:MARKer<x>:Y? (Query Only)

Queries the marker amplitude of the selected marker in the Occupied Bandwidth measurement.

Conditions

Measurement views: Occupied Bandwidth

Group

Calculate commands

Syntax

```
CALCulate:OBWidth:MARKer<x>:Y?
```

Related Commands

CALCulate:OBWidth:MARKer<x>:X (on page 497)

Arguments

None

Returns

`<NRf>` Marker amplitude of the selected marker.

Examples

`CALCULATE:OBWIDTH:MARKER1:Y?` might return -34.28 indicating Marker 1(M1) is at -34.28 dBm.

CALCulate:OFDM:CONSte:MARKer<x>:FREQuency

Sets or queries the frequency position of the selected marker in the OFDM Constellation view.

Conditions

Measurement views: OFDM

Group

Calculate commands

Syntax

```
CALCulate:OFDM:CONSte:MARKer<x>:FREQuency <value>
CALCulate:OFDM:CONSte:MARKer<x>:FREQuency?
```

Arguments

`<value>::=<NRf>` specifies the frequency position of the marker. Range: Start to Stop frequency (left to right edge of the horizontal axis). Using an out-of-range value causes an execution error (-222, "Data out of range").

Returns

`<NRf>` The value type depends on where the marker is placed.

Examples

`CALCulate:OFDM:CONSte:MARKer1:FREQuency?` might return 800 MHz,indicating that Marker 1 M1 is positioned at the 800 MHz position in the OFDM Constellation view.

CALCulate:OFDM:CONSte:MARKer<x>:MAGNitude? (Query Only)

Queries the marker magnitude of the selected marker in the OFDM Constellation view.

Conditions

Measurement views: OFDM

Group

Calculate commands

Syntax

```
CALCulate:OFDM:CONSte:MARKer<x>:MAGNitude?
```

Arguments

None

Returns

`<NRf>` The magnitude for the selected marker in dB.

Examples

`CALCulate:OFDM:CONSte:MARKer1:MAGNitude` might return 35 indicating readout of Marker 1 (M1) is 35 dB.

CALCulate:OFDM:CONSte:MARKer<x>:PHASe? (Query Only)

Queries the phase readout of the selected marker in the OFDM Constellation measurement view.

Conditions

Measurement views: OFDM

Group

Calculate commands

Syntax

```
CALCulate:OFDM:CONSte:MARKer<x>:PHASe?
```

Arguments

None

Returns

<NRf> The phase readout for the selected marker.

Examples

CALCulate:OFDM:CONSte:MARKer1:PHASe? might return 35.74, indicating the phase readout of Marker 1 (M1) is 35.74°.

CALCulate:OFDM:CONSte:MARKer<x>:TIME

Sets or queries the X position of the selected marker in the OFDM Constellation measurement view.

Conditions

Measurement views: OFDM

Group

Calculate commands

Syntax

```
CALCulate:OFDM:CONSte:MARKer<x>:TIME <value>
CALCulate:OFDM:CONSte:MARKer<x>:TIME?
```

Arguments

<value::=NRf> specifies the marker time.

Examples

CALCulate:OFDM:CONSte:MARKer1:TIME –234.5us places Marker 1 (M1) at the –234.5 µs on the trace.

CALCulate:OFDM:CONSte:MARKer<x>:TYPE? (Query Only)

Queries the data type of the selected marker in OFDM Constellation view.

Conditions

Measurement views: OFDM

Group

Calculate commands

Syntax

```
CALCulate:OFDM:CONSte:MARKer<x>:TYPE?
```

Arguments

None

Returns

1.000 if the marker type is a data marker.

0.000 if the marker type is a pilot marker.

Examples

CALCulate:OFDM:CONSte:MARKer1:TYPE? might return 1.000 indicating Marker 1 (M1) is a data marker.

CALCulate:OFDM:CONSte:MARKer<x>:VALue? (Query Only)

Queries the value of the specified marker in OFDM Constellation view.

Conditions

Measurement views: OFDM

Group

Calculate commands

Syntax

CALCulate:OFDM:CONSte:MARKer<x>:VALue?

Arguments

None

Returns

<NRf> The value readout for the selected marker.

Examples

CALCulate:OFDM:CONSte:MARKer2:VALue? might return 2.00, indicating the value readout of Marker 2 (M2) is 2.

CALCulate:OFDM:CRESponse:MARKer<x>:DELTa:X? (Query Only)

Returns the delta marker frequency for the selected marker in the OFDM Channel response view.

Conditions

Measurement views: OFDM

Group

Calculate commands

Syntax

CALCulate:OFDM:CRESponse:MARKer<x>:DELTa:X?

Arguments

None

Returns

<NRf> Delta Marker frequency for the selected marker.

Examples

CALCulate:OFDM:CRESponse:MARKer2:DELTa:X might return 1.28E+6 indicating that the delta marker frequency for Marker 2 (M2) is 1.28 MHz.

CALCulate:OFDM:CRESponse:MARKer<x>:DELTa:Y? (Query Only)

Returns the delta marker frequency for the selected marker in the OFDM Channel response view.

Conditions

Measurement views: OFDM

Group

Calculate commands

Syntax

`CALCulate:OFDM:CRESponse:MARKer<x>:DELTa:Y?`

Arguments

None

Returns

`<NRf>` Delta Marker frequency for the selected marker.

Examples

`CALCulate:OFDM:CRESponse:MARKer2:DELTa:Y` might return 1.28E+6 indicating that the delta marker frequency for Marker 2 (M2) is 1.28 MHz.

CALCulate:OFDM:CRESponse:MARKer<x>:MAXimum (No Query Form)

Moves the selected marker to the highest peak in the OFDM Channel response view.

Conditions

Measurement views: OFDM

Group

Calculate commands

Syntax

`CALCulate:OFDM:CRESponse:MARKer<x>:MAXimum`

Arguments

None

Examples

`CALCulate:OFDM:CRESponse:MARKer1:MAXimum` moves Marker 1 (M1) to the highest peak in the OFDM Channel response view.

CALCulate:OFDM:CRESponse:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the selected marker to the next peak higher in amplitude in the OFDM Channel response view.

Conditions

Measurement views: OFDM

Group

Calculate commands

Syntax

`CALCulate:OFDM:CRESponse:MARKer<x>:PEAK:HIGHer`

Arguments

None

Examples

`CALCulate:OFDM:CRESponse:MARKer1:PEAK:HIGHer` moves Marker 1(M1) to the next peak higher in amplitude in the OFDM Channel response view.

CALCulate:OFDM:CRESponse:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the selected marker to the next peak to the left in the OFDM Channel response view.

Conditions

Measurement views: OFDM

Measurement views: OFDM

Calculate commands

Syntax

`CALCulate:OFDM:CRESponse:MARKer<x>:PEAK:LEFT`

Arguments

None

Examples

`CALCulate:OFDM:CRESponse:MARKer1:PEAK:LEFT` moves Marker1 (M1) to the next peak to the left in the OFDM Channel response view.

CALCulate:OFDM:CRESponse:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the selected marker to the next peak lower in amplitude in the OFDM Channel response view.

Conditions

Measurement views: OFDM

Group

Calculate commands

Syntax

`CALCulate:OFDM:CRESponse:MARKer<x>:PEAK:LOWer`

Arguments

None

Examples

`CALCulate:OFDM:CRESponse:MARKer2:PEAK:LOWer` moves Marker 2 (M2) to the next peak lower in amplitude in the OFDM Channel response view.

CALCulate:OFDM:CRESponse:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the selected marker to the next peak to the right in the OFDM Channel response view.

Conditions

Measurement views: OFDM

Group

Calculate commands

Syntax

`CALCulate:OFDM:CRESponse:MARKer<x>:PEAK:RIGHt`

Arguments

None

Examples

`CALCulate:OFDM:CRESponse:MARKer1:PEAK:RIGHt` moves Marker1 (M1) to the next peak to the right in the OFDM Channel response view.

CALCulate:OFDM:CRESponse:MARKer<x>:X

Sets or queries the horizontal position of the selected marker in the OFDM

Channel response view.

Conditions

Measurement views: OFDM

Group

Calculate commands

Syntax

`CALCulate:OFDM:CRESponse:MARKer<x>:X <value>`
`CALCulate:OFDM:CRESponse:MARKer<x>:X?`

Arguments

`<value::=<NRf>` specifies the horizontal position of the marker. Range: Start to Stop frequency (left to right edge of the horizontal axis). Using an out-of-range value causes an execution error (-222, "Data out of range").

Examples

`CALCulate:OFDM:CRESponse:MARKer1:X 800MHZ` places Marker 1 (M1) at 800 MHz on the trace.

CALCulate:OFDM:CRESponse:MARKer<x>:Y? (Query Only)

Returns the amplitude of the selected marker in the OFDM Channel response view.

Conditions

Measurement views: OFDM

Group

Calculate commands

Syntax

`CALCulate:OFDM:CRESponse:MARKer<x>:Y?`

Arguments

None

Returns

`<NRf>` Amplitude of the selected marker in dB.

Examples

`CALCulate:OFDM:CRESponse:MARKer1:Y` might return 33.393856E–3 indicating the amplitude of Marker 1 (M1) is 0.033 dB.

CALCulate:OFDM:FLATness:MARKer<x>:MAXimum (No Query Form)

Positions the specified marker at the maximum point on the trace in the OFDM Spectral Flatness display.

Conditions

Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group

Calculate commands

Syntax

`CALCulate:OFDM:FLATness:MARKer<x>:MAXimum`

Arguments

None

Examples

`CALC:OFDM:FLAT:MARK2:MAX` positions the Marker 2 (M2) at the maximum point on the trace.

CALCulate:OFDM:FLATness:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the specified marker to the next peak higher in amplitude on the trace on the OFDM Spectral Flatness display.

Conditions

Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group

Calculate commands

Syntax

`CALCulate:OFDM:FLATness:MARKer<x>:PEAK:HIGHer`

Arguments

None

Examples

`CALC:OFDM:FLAT:MARK2:PEAK:HIGHer` moves Marker 2 (M2) to the next peak higher in amplitude on the trace.

CALCulate:OFDM:FLATness:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the specified marker to the next peak to the left on the trace in the OFDM Spectral Flatness display.

Conditions

Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group

Calculate commands

Syntax

`CALCulate:OFDM:FLATness:MARKer<x>:PEAK:LEFT`

Arguments

None

Examples

`CALC:OFDM:FLAT:MARK2:PEAK:LEFT` moves Marker 2 (M2) to the next peak to the left on the trace.

CALCulate:OFDM:FLATness:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the marker to the next peak lower in amplitude on the trace on the OFDM Spectral Flatness display.

Conditions

Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group

Calculate commands

Syntax

`CALCulate:OFDM:FLATness:MARKer<x>:PEAK:LOWer`

Arguments

None

Examples

`CALC:OFDM:FLAT:MARK2:PEAK:LOW` moves Marker 2 (M2) to the next peak lower on the trace.

CALCulate:OFDM:FLATness:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the marker to the next peak to the right on the trace on the OFDM Spectral Flatness display.

Conditions

Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group

Calculate commands

Syntax

```
CALCulate:OFDM:FLATness:MARKer<x>:PEAK:RIGHt
```

Arguments

None

Examples

`CALC:OFDM:FLAT:MARK2:PEAK:RIGHT` moves Marker 2 (M2) to the next peak to the next peak on the right on the trace.

CALCulate:OFDM:FLATness:MARKer<x>:X

Sets or queries the horizontal value at the selected marker position on the points trace in the OFDM Spectral Flatness display. The units are in Frequency or Subcarrier.

To set the units, use the command `[SENSe]:OFDM:UNIT:FREQuency`.

Conditions

Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group

Calculate commands

Syntax

```
CALCulate:OFDM:FLATness:MARKer<x>:X <NRf>
CALCulate:OFDM:FLATness:MARKer<x>:X?
```

Arguments

Floating point number that specifies the horizontal position of the marker. Range:Start to Stop frequency (left to right edge of the horizontal axis). Using an out-of-range value causes an execution error (-222, “Data out of range”).

Examples

`CALC:OFDM:FLAT:MARK3:X 22` places Marker 3 at 22 Subcarrier on the trace(assuming the units have been set to Subcarrier).

CALCulate:OFDM:FLATness:MARKer<x>:Y? (Query Only)

Returns the value of the amplitude (vertical position) at the selected marker position on the OFDM Spectral Flatness display.

Conditions

Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group

Calculate commands

Syntax

`CALCulate:OFDM:FLATness:MARKer<x>:Y?`

Returns

Floating point number that indicates the amplitude at the selected marker position in dB.

Examples

`CALC:OFDM:FLAT:MARK1:Y?` might return 33.393856E-3 indicating the amplitude at Marker 1 is 0.033 dB.

CALCulate:OFDM:TABLe:MARKer<x>:FREQuency

Sets or queries the frequency position of the selected marker in the OFDM Symbol table view.

Conditions

Measurement views: OFDM

Group

Calculate commands

Syntax

```
CALCulate:OFDM:TABLe:MARKer<x>:FREQuency <value>
CALCulate:OFDM:TABLe:MARKer<x>:FREQuency?
```

Arguments

`<value>::=<NRf>` specifies the marker frequency.

Examples

`CALCulate:OFDM:TABLe:MARKer1:FREQuency 800MHZ` places Marker 1(M1) at 800 MHz in the symbol table.

CALCulate:OFDM:TABLe:MARKer<x>:TIMe

Sets or queries the marker time in the OFDM Symbol table view.

Conditions

Measurement views: OFDM

Group

Calculate commands

Syntax

```
CALCulate:OFDM:TABLe:MARKer<x>:TIMe <value>
CALCulate:OFDM:TABLe:MARKer<x>:TIMe?
```

Arguments

`<value>::=<NRf>` specifies the marker time.

Examples

`CALCulate:OFDM:TABLe:MARKer1:TIMe -234.5us` places Marker 1 (M1) at –234.5 µs in the symbol table.

CALCulate:OFDM:TABLe:MARKer<x>:VALue? (Query Only)

Queries the value readout of the selected marker in the OFDM Symbol table view.

Conditions

Measurement views: OFDM

Group

Calculate commands

Syntax

```
CALCulate:OFDM:TABLe:MARKer<x>:VALue?
```

Arguments

None

Returns

`<NRf>` The value readout of the selected marker.

Examples

`CALCulate:OFDM:TABLe:MARKer2:VALue?` might return 2.00 indicating the value readout of Marker 2 (M2) is 2.

CALCulate:P25:CONSte:MARKer<x>:DELTa:X[:TIME]? (Query Only)

Returns the delta marker time for the selected marker on the P25 Constellation display.

Conditions

Measurement view: P25 Constellation

The parameter <x> = 1 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Group

Calculate commands

Syntax

```
CALCulate:P25:CONSte:MARKer<x>:DELTa:X[:TIME]?
```

Returns

<NR2> = a floating point value without an exponent. This is the time for the selected Delta marker.

NOTE: Use the [SENSe]:P25:TIMe:UNITs command to select the time unit: symbols (default) or seconds.

Examples

CALCulate:P25:CONSte:MARKer1:DELTa:X:TIME? might return 62.75,indicating that the Delta marker time is 62.75 symbols.

CALCulate:P25:CONSte:MARKer<x>:FDEViation? (Query Only)

Queries the frequency deviation of the selected marker in the P25 Constellation display.

Conditions

Measurement view: P25 Constellation

The parameter <x> =0 to 4.

The specified marker must be activated using the CALCulate:MARKer:ADD command.

Group

Calculate commands

Syntax

CALCulate:P25:CONSte:MARKer<x>:FDEViation?

Returns

<NR2> = a floating point value without an exponent. This is the frequency deviation readout for the selected marker.

Examples

CALC:P25:CONS:MARr1:FDEV? might return 102.82380000E+3, indicating the frequency deviation readout of Marker 1 (M1) is 102.824 kHz.

CALCulate:P25:CONSte:MARKer<x>:MAGNitude? (Query Only)

Queries the frequency deviation (for Freq Dev trace types) or magnitude (for IQ trace types) of the selected marker in the P25 Constellation display.

Conditions

Measurement view: P25 Constellation

Group

Calculate commands

Syntax

CALCulate:P25:CONSte:MARKer<x>:MAGNitude?

Returns

<NRf> is the frequency deviation or magnitude of the selected marker.

Examples

CALC:P25:CONS:MARK1:MAGN? might return 1.6257750459E+3 if the trace type is Freq Dev, indicating that Marker 1 (M1) has a frequency deviation of 1.626 kHz.

CALC:P25:CONS:MARK1:MAGN? might return 1.0048475788 if the trace type is IQ, indicating that Marker1 (M1) has a magnitude of 1.005.

CALCulate:P25:CONSte:MARKer<x>:MAXimum (No Query Form)

Moves the selected marker to the highest peak on the trace in the P25 Constellation display.

Conditions

Measurement view: P25 Constellation

The specified marker must be activated using the CALCulate:MARKer:ADD command.

Group

Calculate commands

Syntax

CALCulate:P25:CONSte:MARKer<x>:MAXimum

Related Commands

CALCulate:P25:CONSte:MARKer<x>:PEAK:LEFT (on page 511)

CALCulate:P25:CONSte:MARKer<x>:PEAK:RIGHt (on page 511)

Examples

CALC:P25:CONS:MARK1:MAX positions Marker 1 (M1) at the symbol in the center of the time record.

CALCulate:P25:CONSte:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the selected marker to the next peak to the left on the trace in the P25 Constellation display.

Conditions

Measurement view: P25 Constellation

Group

Calculate commands

Syntax

CALCulate:P25:CONSte:MARKer<x>:PEAK:LEFT

Related Commands

CALCulate:P25:CONSte:MARKer<x>:PEAK:RIGHt (on page 511)

CALCulate:P25:CONSte:MARKer<x>:MAXimum (on page 511)

Examples

CALC:P25:CONS:MARK1:PEAK:LEFT moves Marker 1 (M1) in the time domain to the next lower symbol number.

CALCulate:P25:CONSte:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the selected marker to the next peak to the right on the trace in the P25 Constellation display.

Conditions

Measurement view: P25 Constellation

The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Group

Calculate commands

Syntax

`CALCulate:P25:CONSte:MARKer<x>:PEAK:RIGHt`

Related Commands

CALCulate:P25:CONSte:MARKer<x>:MAXimum (on page 511)

CALCulate:P25:CONSte:MARKer<x>:PEAK:LEFT (on page 511)

Examples

`CALC:P25:CONS:MARK1:PEAK:RIGH` moves the Marker 1 (M1) in the time domain to the next higher symbol number to the right.

CALCulate:P25:CONSte:MARKer<x>:PHASe? (Query Only)

Queries the phase of the selected marker in the P25 Constellation display.

Conditions

Measurement view: P25 Constellation

IQ trace type required.

Group

Calculate commands

Syntax

`CALCulate:P25:CONSte:MARKer<x>:PHASe?`

Returns

<NR2> = a floating point value without an exponent. This is the phase value for the selected marker.

Examples

`CALC:P25:CONS:MARK1:PHAS?` might return 41.4721108251, indicating the phase of Marker 1 (M1) is 41.5°.

CALCulate:P25:CONSte:MARKer<x>:SYMBol? (Query Only)

Queries the readout symbol of the selected marker in the P25 Constellation display.

Conditions

Measurement view: P25 Constellation

The parameter <x> = 0,1,2,3,4 is the marker.

The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Group

Calculate commands

Syntax

```
CALCulate:P25:CONSte:MARKer<x>:SYMBol?
```

Returns

<NR2> = a floating point value is the readout symbol for the selected marker.

Examples

`CALCulate:P25:CONSte:MARKer1:SYMBol?` might return 62.0000000000,indicating the readout symbol of Marker 1 (M1) is 62.

CALCulate:P25:CONSte:MARKer<x>:Trace

Places the selected marker on the Demodulated I and Q versus Time trace. The query returns the name of the trace on which the marker resides.

Conditions

Measurement view: P25 Constellation

The parameter <x>= 0 to 4.

The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Group

Calculate commands

Syntax

```
CALCulate:P25:CONSte:MARKer<x>:Trace {Trace1 | Trace2}
CALCulate:P25:CONSte:MARKer<x>:Trace?
```

Arguments

`Trace 1`: sets the I trace marker on.

`Trace 2`: sets the Q trace marker on.

Returns

Trace 1 = I trace

Trace 2 = Q trace

Examples

`CALC:P25:CONS:MARK1:TRAC` Trace1 places Marker 1 (M1) on the I trace.

CALCulate:P25:CONSte:MARKer<x>:VALue? (Query Only)

Queries the readout value of the specified marker in P25 Constellation display.

Conditions

Measurement view: P25 Constellation

Group

Calculate commands

Syntax

```
CALCulate:P25:CONSte:MARKer<x>:VALue?
```

Returns

<NRf> is the readout value for the selected marker.

Examples

CALC:P25:CONS:MARK2:VAL? might return 2.00, indicating the readout value of Marker 2 (M2) is 2.

CALCulate:P25:CONSte:MARKer<x>:X

Sets or queries the horizontal position of the selected marker in the P25 Constellation display.

Conditions

Measurement view: P25 Constellation

Group

Calculate commands

Syntax

```
CALCulate:P25:CONSte:MARKer<x>:X <value>
CALCulate:P25:CONSte:MARKer<x>:X?
```

Arguments

<value>::=<NRf> specifies the time position of the marker. Range is (analysis offset) to [(analysis offset) + (analysis length)]

Returns

Offset set value(-500Hz t0 +500Hz).

Examples

CALCulate:P25:PVTime:MARKer1:X 800MHz places Marker 1 (M1) at 800 MHz on the trace. When used as a query, the return will show the horizontal position of the trace. For example, 800 MHz.

CALCulate:P25:EDIagram:MARKer<x>:DELTa:X[:TIME]? (Query Only)

Returns the delta marker time for the selected marker on the P25 Eye Diagram trace.

Conditions

Measurement view: P25 Eye Diagram

The parameter <x>= 1 to 4. MARKer0 (reference marker) is invalid.

The specified marker must be activated using the CALCulate:MARKer:ADD command.

Group

Calculate commands

Syntax

```
CALCulate:P25:EDIagram:MARKer<x>:DELTa:X[:TIME]?
```

Related Commands

CALCulate:P25:EDIagram:MARKer<x>:DELTa:Y? (on page 515)

Returns

<NRf> is the Delta marker time for the selected marker. Use the [SENSe]:P25:TIMe:UNITs command to select the time unit: symbols (default) or seconds.

Examples

CALCulate:P25:EDIagram:MARKer1:DELTa:X:TIME? might return 62.750000000, Indicating that the delta marker time is 62.75 symbols.

CALCulate:P25:EDIagram:MARKer<x>:DELTa:Y? (Query Only)

Returns the delta marker amplitude for the selected marker on the P25 Eye Diagram trace.

Conditions

Measurement view: P25 Eye Diagram

The parameter <x>= 1 to 4.

The specified marker must be activated using the CALCulate:MARKer:ADD command.

Group

Calculate commands

Syntax

CALCulate:P25:EDIagram:MARKer<x>:DELTa:Y?

Related Commands

CALCulate:P25:EDIagram:MARKer<x>:DELTa:X[:TIME]? (Query Only) (on page 514)

Returns

<NRf> is the Delta marker amplitude for the selected marker.

Examples

CALCulate:P25:EDIagram:MARKer1:DELTa:Y? might return -1.04300000,indicating that the delta marker amplitude is -1.043.

CALCulate:P25:EDIagram:MARKer<x>:MAXimum (No Query Form)

Moves the selected marker to the highest peak on the P25 Eye Diagram trace.

Conditions

Measurement view: P25 Eye Diagram

The parameter <x>=0 to 4.

The specified marker must be activated using the CALCulate:MARKer:ADD command.

Group

Calculate commands

Syntax

```
CALCulate:P25:EDIagram:MARKer<x>:MAXimum
```

Related Commands

CALCulate:P25:EDIagram:MARKer<x>:PEAK:HIGHer (on page 516)

CALCulate:P25:EDIagram:MARKer<x>:PEAK:LEFT (on page 517)

CALCulate:P25:EDIagram:MARKer<x>:PEAK:LOWer (on page 517)

CALCulate:P25:EDIagram:MARKer<x>:PEAK:RIGHt (on page 518)

Arguments

None

Returns

None

Examples

`CALCulate:P25:EDIagram:MARKer1:MAXImum` moves Marker 1 (M1) to the highest peak on the trace.

CALCulate:P25:EDIagram:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the selected marker to the next peak higher in amplitude on the P25 Eye Diagram trace.

Conditions

Measurement view: P25 Eye Diagram

The parameter <x>=0 to 4.

The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Group

Calculate commands

Syntax

```
CALCulate:P25:EDIagram:MARKer<x>:PEAK:HIGHer
```

Related Commands

CALCulate:P25:EDIagram:MARKer<x>:MAXimum (on page 515)

CALCulate:P25:EDIagram:MARKer<x>:PEAK:LEFT (on page 517)

CALCulate:P25:EDIagram:MARKer<x>:PEAK:LOWer (on page 517)

CALCulate:P25:EDIagram:MARKer<x>:PEAK:RIGHt (on page 518)

Arguments

None

Returns

None

Examples

`CALCulate:P25:EDIagram:MARKer1:PEAK:HIGHer` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

CALCulate:P25:EDIagram:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the selected marker to the next peak to the left on the P25 Eye Diagram trace.

Conditions

Measurement view: P25 Eye Diagram

The parameter <x>=0 to 4.

The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Group

Calculate commands

Syntax

`CALCulate:P25:EDIagram:MARKer<x>:PEAK:LEFT`

Related Commands

CALCulate:P25:EDIagram:MARKer<x>:MAXimum (on page 515)

CALCulate:P25:EDIagram:MARKer<x>:PEAK:HIGHer (on page 516)

CALCulate:P25:EDIagram:MARKer<x>:PEAK:LOWer (on page 517)

CALCulate:P25:EDIagram:MARKer<x>:PEAK:RIGHt (on page 518)

Arguments

None

Returns

None

Examples

`CALCulate:P25:EDIagram:MARKer1:PEAK:LEFT` moves Marker 1 (M1) to the next peak to the left on the trace.

CALCulate:P25:EDIagram:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the selected marker to the next peak lower in amplitude on the P25 Eye Diagram trace.

Conditions

Measurement view: P25 Eye Diagram

The parameter <x>=0 to 4.

The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Group

Calculate commands

Syntax

`CALCulate:P25:EDIagram:MARKer<x>:PEAK:LOWer`

Related Commands

CALCulate:P25:EDIagram:MARKer<x>:MAXimum (on page 515)

CALCulate:P25:EDIagram:MARKer<x>:PEAK:HIGHer (on page 516)

CALCulate:P25:EDIagram:MARKer<x>:PEAK:LEFT (on page 517)

CALCulate:P25:EDIagram:MARKer<x>:PEAK:RIGHt (on page 518)

Arguments

None

Returns

None

Examples

`CALCulate:P25:EDIagram:MARKer1:PEAK:LOWer` moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

CALCulate:P25:EDIagram:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the selected marker to the next peak to the right on the P25 Eye Diagram trace.

Conditions

Measurement view: P25 Eye Diagram

The parameter <x>=0 to 4.

The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Group

Calculate commands

Syntax

`CALCulate:P25:EDIagram:MARKer<x>:PEAK:RIGHt`

Related Commands

CALCulate:P25:EDIagram:MARKer<x>:MAXimum (on page 515)

CALCulate:P25:EDIagram:MARKer<x>:PEAK:HIGHer (on page 516)

CALCulate:P25:EDIagram:MARKer<x>:PEAK:LEFT (on page 517)

CALCulate:P25:EDIagram:MARKer<x>:PEAK:LOWer (on page 517)

Arguments

None

Returns

None

Examples

`CALCulate:P25:EDIagram:MARKer1:PEAK:RIGHt` moves Marker 1 (M1) to the next peak to the right on the trace.

CALCulate:P25:EDIagram:MARKer<x>:TRACe

Sets the selected marker on the I or Q trace, or queries the trace that the specified marker is on, in the P25 Eye Diagram display.

Conditions

Measurement view: P25 Eye Diagram

The parameter <x>=0 to 4.

The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Group

Calculate commands

Syntax

```
CALCulate:P25:EDIagram:MARKer<x>:TRACe { TRACe1 | TRACe2 }
CALCulate:P25:EDIagram:MARKer<x>:TRACe?
```

Arguments

`TRACe1` places the specified marker on the I trace.

`TRACe2` places the specified marker on the Q trace.

Returns

TRACe1 means the specified marker is on the I trace.

TRACe2 means the specified marker is on the Q trace.

Examples

`CALCulate:P25:EDIagram:MARKer1:TRACe` TRACe1 places Marker 1 (M1) on the I trace in the P25 Eye Diagram display.

CALCulate:P25:EDIagram:MARKer<x>:X

Sets or queries the horizontal position (time) of the selected marker in the P25 Eye Diagram display.

Conditions

Measurement view: P25 Eye Diagram

The parameter <x>=0 to 4.

The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Group

Calculate commands

Syntax

```
CALCulate:P25:EDIagram:MARKer<x>:X <NRf>
CALCulate:P25:EDIagram:MARKer<x>:X?
```

Related Commands

CALCulate:P25:EDIagram:MARKer<x>:Y (on page 520)

Arguments

<NRf> specifies the horizontal position (time) of the marker.

NOTE: Use the [SENSe]:P25:TIMe:UNITs command to select the time unit:Symbols (default) or seconds.

Returns

<NRf> is the horizontal position of the selected marker.

Examples

CALC:P25:EDI:MARK1:X 38.5 places Marker 1 (M1) at 38.5 symbols on the trace.

CALCulate:P25:EDIagram:MARKer<x>:Y

Sets or queries the vertical position of the selected marker in the P25 Eye Diagram display.

Conditions

Measurement view: P25 Eye Diagram

The parameter <x>=0 to 4.

The specified marker must be activated using the CALCulate:MARKer:ADD command.

Group

Calculate commands

Syntax

```
CALCulate:P25:EDIagram:MARKer<x>:Y <value>
CALCulate:P25:EDIagram:MARKer<x>:Y?
```

Related Commands

CALCulate:P25:EDIagram:MARKer<x>:X (on page 519)

Arguments

<value>:: =<NRf> specifies the vertical position of the marker.

Returns

<NRf> is the vertical position of the selected marker.

Examples

CALC:P25:EDI:MARK1:Y? might return 571.8E-3, indicating Marker 1 (M1) is at 0.5718.

CALCulate:P25:PVTime:MARKer<x>:DELTa:X[:TIME]? (Query Only)

Returns the delta marker time (seconds) for the selected marker on the Power vs. Time display.

Conditions

Measurement view: P25 Power vs. Time

The parameter <x> = 1,2,3,4 is the marker.

The specified marker must be activated using the CALCulate:MARKer:ADD command.

Group

Calculate commands

Syntax

```
CALCulate:P25:PVTime:MARKer<x>:DELTa:X[:TIME]?
```

Related Commands

CALCulate:P25:PVTime:MARKer<x>:DELTa:Y[:TIME]? (on page 521)

SENSe:P25:TIMe:UNITs (on page 2248)

Returns

<NRf> is the Delta marker time for the selected marker.

NOTE: Use the SENSe:P25:TIME:UNITs command to select the time unit symbols (default) or seconds.

Examples

CALC:P25:PVT:MARK1:DELT:X? might return -3.99372800000E-3, indicating that the delta marker 1 (M1) time is -3.937 ms.

CALCulate:P25:PVTime:MARKer<x>:DELTa:Y[:TIME]? (Query Only)

Returns the Delta marker amplitude (dB) for the selected marker on the Power vs. Time display.

Conditions

Measurement view: P25 Power vs. Time

The parameter <x> = 1,2,3,4 is the marker.

The specified marker must be activated using the CALCulate:MARKer:ADD command.

Group

Calculate commands

Syntax

```
CALCulate:P25:PVTime:MARKer<x>:DELTa:Y[:TIME]?
```

Related Commands

CALCulate:P25:PVTime:MARKer<x>:DELTa:X[:TIME]? (on page 520)

Returns

<NR3> is a floating point value with an exponent. This is the Delta marker amplitude.

Examples

CALC:P25:PVT:MARK1:DELT:Y? might return 552.6123046875E-3, indicating that the Delta marker 1 (M1) amplitude is 0.55 dB.

CALCulate:P25:PVTime:MARKer<x>:Maximum (No Query Form)

Moves the selected marker to the highest peak on the trace in the Power vs. Time display.

Conditions

Measurement view: P25 Power vs. Time

Group

Calculate commands

Syntax

```
CALCulate:P25:PVTime:MARKer<x>:Maximum
```

Related Commands

CALCulate:P25:PVTime:MARKer<x>:PEAK:LOWer (on page 523)

CALCulate:P25:PVTime:MARKer<x>:PEAK:HIGHer (on page 522)

CALCulate:P25:PVTime:MARKer<x>:PEAK:LEFT (on page 523)

CALCulate:P25:PVTime:MARKer<x>:PEAK:RIGHt (on page 524)

Arguments

None

Returns

None

Examples

`CALC:P25:PVT:MARK1:PEAK:MAX` moves Marker 1 (M1) to the highest peak on the trace.

CALCulate:P25:PVTime:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the selected marker to the next peak higher in amplitude on the trace in the Power vs. Time display.

Conditions

Measurement view: P25 Power vs. Time

Group

Calculate commands

Syntax

```
CALCulate:P25:PVTime:MARKer<x>:PEAK:HIGHer
```

Related Commands

CALCulate:P25:PVTime:MARKer<x>:Maximum (on page 521)

CALCulate:P25:PVTime:MARKer<x>:PEAK:LOWer (on page 523)

CALCulate:P25:PVTime:MARKer<x>:PEAK:LEFT (on page 523)

CALCulate:P25:PVTime:MARKer<x>:PEAK:RIGHt (on page 524)

Arguments

None

Returns

None

Examples

`CALC:P25:PVT:MARK1:PEAK:HIGH` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

CALCulate:P25:PVTime:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the selected marker to the next peak to the left on the trace in the Power vs. Time display.

Conditions

Measurement view: P25 Power vs. Time

Group

Calculate commands

Syntax

`CALCulate:P25:PVTime:MARKer<x>:PEAK:LEFT`

Related Commands

CALCulate:P25:PVTime:MARKer<x>:PEAK:LOWer (on page 523)

CALCulate:P25:PVTime:MARKer<x>:Maximum (on page 521)

CALCulate:P25:PVTime:MARKer<x>:PEAK:HIGHer (on page 522)

CALCulate:P25:PVTime:MARKer<x>:PEAK:RIGHt (on page 524)

Arguments

None

Returns

None

Examples

`CALC:P25:PVT:MARK1:PEAK:LEFT` moves Marker 1 (M1) to the next peak to the left on the trace.

CALCulate:P25:PVTime:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the selected marker to the next peak lower in amplitude on the trace in the Power vs. Time display.

Conditions

Measurement view: P25 Power vs. Time

Group

Calculate commands

Syntax

`CALCulate:P25:PVTime:MARKer<x>:PEAK:LOWer`

Related Commands

CALCulate:P25:PVTime:MARKer<x>:PEAK:HIGHer (on page 522)

CALCulate:P25:PVTime:MARKer<x>:Maximum (on page 521)

CALCulate:P25:PVTime:MARKer<x>:PEAK:LEFT (on page 523)

CALCulate:P25:PVTime:MARKer<x>:PEAK:RIGHt (on page 524)

Arguments

None

Returns

None

Examples

`CALC:P25:PVT:MARK1:PEAK:LOW` moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

CALCulate:P25:PVTime:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the selected marker to the next peak to the right on the trace in the Power vs. Time display.

Conditions

Measurement view: Power vs. Time

Group

Calculate commands

Syntax

`CALCulate:P25:PVTime:MARKer<x>:PEAK:RIGHt`

Related Commands

CALCulate:P25:PVTime:MARKer<x>:PEAK:LOWer (on page 523)

CALCulate:P25:PVTime:MARKer<x>:PEAK:HIGHer (on page 522)

CALCulate:P25:PVTime:MARKer<x>:Maximum (on page 521)

CALCulate:P25:PVTime:MARKer<x>:PEAK:LEFT (on page 523)

Arguments

None

Returns

None

Examples

`CALC:P25:PVT:MARK1:PEAK:RIGH` moves Marker 1 (M1) to the next peak to the right on the trace.

CALCulate:P25:PVTime:MARKer<x>:X

Sets or queries the horizontal position of the selected marker in the Power vs.Time display.

Conditions

Measurement view: P25 Power vs. Time

Group

Calculate commands

Syntax

```
CALCulate:P25:PVTime:MARKer<x>:X <value>
CALCulate:P25:PVTime:MARKer<x>:X?
```

Related Commands

CALCulate:P25:PVTime:MARKer<x>:Y (on page 525)

Arguments

`<value>::=<NRf>` specifies the horizontal position of the marker. Range is Start to Stop frequency (left to right edge of the horizontal axis). Using an out-of-range value causes an execution error (-222, "Data out of range").

Returns

`<value>::=<NRf>` specifies the horizontal position of the marker.

Examples

`CALC:P25:PVT:MARK1:X 800MHz` places Marker 1 (M1) at 800 MHz on the trace. When used as a query, the return will show the horizontal position of the trace. For example, 800 MHz.

CALCulate:P25:PVTime:MARKer<x>:Y

Sets or queries the vertical position of the selected marker in the P25 Power vs.Time display.

Conditions

Measurement view: P25 Power vs. Time

Group

Calculate commands

Syntax

```
CALCulate:P25:PVTime:MARKer<x>:Y <value>
CALCulate:P25:PVTime:MARKer<x>:Y?
```

Related Commands

CALCulate:P25:PVTime:MARKer<x>:X (on page 524)

Arguments

`<value>::=<NRf>` specifies the vertical position of the marker

Returns

`<NR2>` = floating point value without an exponent.

Examples

`CALC:P25:PVT:MARK1:Y?` might return -67.1001052856, indicating that the horizontal position of Marker 1 (M1) is -67.10 dBm.

`CALC:P25:PVT:MARK1:Y -75.66dBm` places Marker 1 (M1) at -75.66 on the trace.

CALCulate:PERRor[:MEASview<y>]:MARKer<x>:DELTa:X? (Query only)

Returns the delta marker time for the selected marker in the Phase error versus Time measurement.

The parameter <x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Phase error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Phase error versus Time).

Group

Calculate commands

Syntax

```
CALCulate:PERRor[:MEASview<y>]:MARKer<x>:DELTa:X?
```

Related Commands

CALCulate:PERRor[:MEASview<y>]:MARKer<x>:DELTa:Y? (on page 526)

Arguments

None

Returns

<NRf> Delta marker time for the selected marker in percent (%).

Examples

CALCulate:PERRor:MEASview2:MARKer1:DELTa:X? might return 3.84, indicating that the delta marker duration is 3.84 for measurement view 2.

CALCulate:PERRor[:MEASview<y>]:MARKer<x>:DELTa:Y? (Query only)

Returns the delta marker phase for the selected marker in the Phase error versus Time measurement.

The parameter <x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Phase error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Phase error versus Time).

Group

Calculate commands

Syntax

```
CALCulate:PERRor[:MEASview<y>]:MARKer<x>:DELTa:Y?
```

Related Commands

CALCulate:PERRor [:MEASview<y>]:MARKer<x>:DELTa:X? (on page 525)

Arguments

None

Returns

<NRf> Delta marker phase for the selected marker in percent (%).

Examples

CALCulate:PERRor:MEASview2:MARKer1:DELTa:Y? might return 3.84, indicating that the delta marker amplitude is 3.84% for measurement view2.

CALCulate:PERRor[:MEASview<y>]:MARKer<x>:MAXimum (No query form)

Moves the selected marker to the highest peak on the trace in the Phase error versus Time measurement.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Phase error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Phase error versus Time).

Group

Calculate commands

Syntax

CALCulate:PERRor[:MEASview<y>]:MARKer<x>:MAXimum

Arguments

None

Examples

CALCulate:PERRor:MEASview2:MARKer1:MAXimum moves Marker 1 (M1) to the highest peak on the trace for measurement view2.

CALCulate:PERRor[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (No query form)

Moves the selected marker to the next peak higher in amplitude on the Phase error versus Time trace.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Phase error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Phase error versus Time).

Group

Calculate commands

Syntax

```
CALCulate:PERRor[:MEASview<y>]:MARKer<x>:PEAK:HIGHer
```

Related Commands

CALCulate:PERRor[:MEASview<y>] :MARKer<x>:PEAK:LOWer (on page 528)

Arguments

None

Examples

`CALCulate:PERRor:MEASview2:MARKer1:PEAK:HIGHer` moves Marker 1 (M1) to the next peak higher in amplitude on the trace for measurement view 2.

CALCulate:PERRor[:MEASview<y>]:MARKer<x>:PEAK:LEFT (No query form)

Moves the selected marker to the next peak to the left on the trace in the Phase error versus Time measurement.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Phase error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Phase error versus Time).

Group

Calculate commands

Syntax

```
CALCulate:PERRor[:MEASview<y>]:MARKer<x>:PEAK:LEFT
```

Related Commands

CALCulate:PERRor[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (on page 529)

Arguments

None

Examples

`CALCulate:PERRor:MEASview2:MARKer1:PEAK:LEFT` moves Marker 1 (M1) to the next peak to the left on the trace for measurement view 2.

CALCulate:PERRor[:MEASview<y>]:MARKer<x>:PEAK:LOWer (No query form)

Moves the selected marker to the next peak lower in amplitude on the Phase error versus Time trace.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Phase error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Phase error versus Time).

Group

Calculate commands

Syntax

`CALCulate:PERRor[:MEASview<y>]:MARKer<x>:PEAK:LOWer`

Related Commands

CALCulate:PERRor[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (on page 527)

Arguments

None

Examples

`CALCulate:PERRor:MEASview2:MARKer1:PEAK:LOWer` moves Marker 1 (M1) to the next peak lower in amplitude on the trace for measurement view 2.

CALCulate:PERRor[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (No query form)

Moves the selected marker to the next peak to the right on the trace in the Phase error versus Time measurement.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Phase error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Phase error versus Time).

Group

Calculate commands

Syntax

`CALCulate:PERRor[:MEASview<y>]:MARKer<x>:PEAK:RIGHt`

Related Commands

CALCulate:PERRor[:MEASview<y>]:MARKer<x>:PEAK:LEFT (on page 528)

Arguments

None

Examples

`CALCulate:PERRor:MEASview2:MARKer1:PEAK:RIGHt` moves Marker 1 (M1) to the next peak to the right on the trace for measurement view 2.

CALCulate:PERRor[:MEASview<y>]:MARKer<x>:X

Sets or queries the horizontal position of the selected marker in the Phase error versus Time measurement.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Phase error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Phase error versus Time).

Group

Calculate commands

Syntax

```
CALCulate:PERRor[:MEASview<y>]:MARKer<x>:X <value>
CALCulate:PERRor[:MEASview<y>]:MARKer<x>:X?
```

Arguments

`<value>::=<NRf>` specifies the horizontal position of the marker. Range: (analysis offset) to [(analysis offset) + (analysis length)]. The unit can be changed by `[SENSe]:DDEMod:TIMe:UNITs` command.

Examples

`CALCulate:PERRor:MEASview2:MARKer1:X 1.5 µs` places Marker 1 (M1) at 1.5 µs on the trace for measurement view 2.

CALCulate:PERRor[:MEASview<y>]:MARKer<x>:Y? (Query only)

Queries the marker amplitude of the selected marker in the Phase error versus Time measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Phase error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Phase error versus Time).

Group

Calculate commands

Syntax

```
CALCulate:PERRor[:MEASview<y>]:MARKer<x>:Y?
```

Related Commands

CALCulate:PERRor[:MEASview<y>]:MARKer<x>:X (on page 530)

Arguments

None

Returns

<NRf> Marker amplitude of the selected marker in percent (%).

Examples

CALCulate:PERRor:MEASview2:MARKer1:Y? might return 5.34, indicating Marker 1 (M1) is at 5.34% for measurement view2.

CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:DELTa:X? (Query Only)

Returns the delta marker time for the selected marker in the Phase versus Time measurement.

Conditions

Measurement views: Phase versus Time

Group

Calculate commands

Syntax

CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:DELTa:X?

Related Commands

CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:DELTa:Y? (on page 531)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Returns

<NRf> Delta marker time for the selected marker.

Examples

CALCULATE:PHVTIME:MEASview1:MARKER1:DELTA:X? might return 38.0E-9, indicating that the delta marker time is 38.0 ns.

CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:DELTa:Y? (Query Only)

Returns the delta marker phase for the selected marker in the Phase versus Time measurement.

Conditions

Measurement views: Phase versus Time

Group

Calculate commands

Syntax

```
CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:DELTa:Y?
```

Related Commands

CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:DELTa:X? (on page 531)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Returns

<NRf> Delta marker phase for the selected marker in degrees.

Examples

CALCULATE:PHVTIME:MEASview1:MARKER1:DELTA:Y? might return 162.38, indicating that the delta marker phase is 162.38 °.

CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:MAXimum (No Query Form)

Moves the selected marker to the highest peak on the trace in the Phase versus Time measurement.

Conditions

Measurement views: Phase versus Time

Group

Calculate commands

Syntax

```
CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:MAXimum
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<x> = 0 to 4.

Examples

CALCULATE:PHVTIME:MEASview1:MARKER1:MAXIMUM moves Marker 1 (M1) to the highest peak on the trace.

CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the selected marker to the next peak higher in amplitude on the Phase versus Time trace.

Conditions

Measurement views: Phase versus Time

Group

Calculate commands

Syntax

```
CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:PEAK:HIGHer
```

Related Commands

CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:PEAK:LOWer (on page 533)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<x> = 0 to 4.

Examples

`CALCULATE:PHVTIME:MEASview1:MARKER1:PEAK:HIGHER` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the selected marker to the next peak to the left on the trace in the Phase versus Time measurement.

Conditions

Measurement views: Phase versus Time

Group

Calculate commands

Syntax

```
CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:PEAK:LEFT
```

Related Commands

CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (on page 534)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<x> = 0 to 4.

Examples

`CALCULATE:PHVTIME:MEASview1:MARKER1:PEAK:LEFT` moves Marker 1 (M1) to the next peak to the left on the trace.

CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the selected marker to the next peak lower in amplitude on the Phase versus Time trace.

Conditions

Measurement views: Phase versus Time

Group

Calculate commands

Syntax

```
CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:PEAK:LOWer
```

Related Commands

CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (on page 532)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<x> = 0 to 4.

Examples

`CALCULATE:PHVTIME:MEASview1:MARKER1:PEAK:LOWER` moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the selected marker to the next peak to the right on the trace in the Phase versus Time measurement.

Conditions

Measurement views: Phase versus Time

Group

Calculate commands

Syntax

```
CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:PEAK:RIGHt
```

Related Commands

CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:PEAK:LEFT (on page 533)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<x> = 0 to 4.

Examples

`CALCULATE:PHVTIME:MEASview1:MARKER1:PEAK:RIGHT` moves Marker 1 (M1) to the next peak to the right on the trace.

CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:TRACe

Sets or queries the trace on which the specified marker is placed in the Phase versus Time measurement.

Conditions

Measurement views: Phase versus Time with specified Measurement view number.

Group

Calculate commands

Syntax

`CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:TRACe {TRACE1 | TRACE5 | TRACE6 | TRACE7}`

`CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:TRACe?`

Arguments

`TRACE1` places the selected marker on Trace 1.

`TRACE5` places the selected marker on Trace 5 (Bitmap trace).

`TRACE6` places the selected marker on Trace 6 (Ogram).

`TRACE7` places marker on Trace 7 (Ogram line).

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`CALCULATE:PHVTIME:MEASVIEW1:MARKER1:TRACe TRACE1` places Marker 1 (M1) on Trace 1 of Phase versus Time Measurement View 1.

CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:X

Sets or queries the horizontal position of the selected marker in the Phase versus Time measurement.

Conditions

Measurement views: Phase versus Time

Group

Calculate commands

Syntax

```
CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:X <value>
CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:X?
```

Related Commands

CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:Y? (on page 536)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<x> = 0 to 4.

<value>::=<NRf> specifies the horizontal position of the marker.Range: (analysis offset) to [(analysis offset) + (analysis length)].

Examples

CALCULATE:PHVTIME:MEASview1:MARKER1:X 1.5US places Marker 1 (M1) at 1.5 µs on the trace.

CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:Y? (Query Only)

Queries the marker amplitude of the selected marker in the Phase versus Time measurement.

Conditions

Measurement views: Phase versus Time

Group

Calculate commands

Syntax

CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:Y?

Related Commands

CALCulate:PHVTime[:MEASview<y>]:MARKer<x>:X (on page 535)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<x> = 0 to 4.

Returns

<NRf> Marker amplitude of the selected marker in degrees.

Examples

CALCULATE:PHVTIME:MEASview1:MARKER1:Y? might return -18.435, indicating Marker 1(M1) is at -18.435 °.

CALCulate:PULSe:CUMulative:HISTogram[:MEASview<y>]:INDicator:X

Sets or queries the horizontal position of the indicator on the trace in the Cumulative Histogram display. Where <x> parameters: 0 to 4

Conditions

Measurement views: Pulse Cumulative Histogram with specified Measurement view number

Group

Calculate commands

Syntax

CALCulate:PULSe:CUMulative:HISTogram[:MEASview<y>]:INDicator:X <value>
CALCulate:PULSe:CUMulative:HISTogram[:MEASview<y>]:INDicator:X?

Arguments

<value>::=<NRf> specifies the horizontal position of the marker.

The parameter <x> = 0 to 4.

Returns

<NRf>

Examples

CALCulate:PULSe:CUMulative:HISTogram:MEASview1:INDicator:X 9.5409090173E-6 places the indicator at the nearest bins mid value at 9.540E-6 on the trace.

CALCulate:PULSe:CUMulative:HISTogram[:MEASview<y>]:INDicator:Y (Query Only)

Queries the vertical position value of the indicator on the trace in the Cumulative Histogram display.

Conditions

Measurement views: Pulse Cumulative Histogram with specified Measurement view number

Group

Calculate commands

Syntax

CALCulate:PULSe:CUMulative:HISTogram[:MEASview<y>]:INDicator:Y?

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<value>

Where <value>::<NRf> is the vertical position value of the marker.

Examples

CALCulate:PULSe:CUMulative:HISTogram:MEASview1:INDicator:Y might return 1.000000 for the vertical position of the marker in cumulative histogram Measurement View 1.

CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:FVTime:DELTa:X (Query Only)

Returns the delta marker time for the selected marker in the Frequency vs Time display of Pulse-Ogram.

Conditions

Measurement views: Pulse-Ogram

Group

Calculate commands

Syntax

CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:FVTime:DELTa:X?

Arguments

<x> = 1, 2, 3, 4 is the marker.The specified marker must be activated using the CALCulate:MARKer:ADD command.

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Returns

<NRf> Delta marker time for the selected marker.

Examples

CALCulate:PULSe:OGRAm:MEASview1:MARKer1:FVTime:DELTa:X? might return 38.0E-6, indicating that the delta marker time for M1 is 38.0 µs.

CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:FVTime:DELTa:Y (Query Only)

Returns the delta marker amplitude for the selected marker in the Frequency vs Time display of Pulse-Ogram.

Conditions

Measurement views: Pulse-Ogram

Group

Calculate commands

Syntax

CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:FVTime:DELTa:Y?

Arguments

<x> = 1, 2, 3, 4 is the marker. The specified marker must be activated using the CALCulate:MARKer:ADD command.

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Returns

<NRf> Delta marker Amplitude for the selected marker

Examples

CALCulate:PULSe:OGRAm:MEASview1:MARKer1:FVTime:DELTa:Y? might return 45.82,indicating that the delta marker amplitude for Marker 1 (M1) is 45.82.

CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:FVTime:MAXimum (No Query Form)

Moves the selected marker to the highest peak on the trace in Frequency vs Time display of Pulse-Ogram.

Conditions

Measurement views: Pulse-Ogram

Group

Calculate commands

Syntax

CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:FVTime:MAXimum

Arguments

<x> = 0,1,2,3,4 is the marker.. MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Examples

CALCulate:PULSe:OGRAm:MEASview1:MARKer1:FVTime:MAXimum moves Marker 1(M1) to the highest peak on the trace.

CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:FVTime:X

Sets or Queries the horizontal position of the selected marker in Frequency vs Time display of Pulse-Ogram.

Conditions

Measurement views: Pulse-Ogram

Group

Calculate commands

Syntax

```
CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:FVTime:X <value>
CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:FVTime:X?
```

Arguments

<x> = 0, 1, 2, 3, 4 is the marker. The specified marker must be activated using the CALCulate:MARKer:ADD command.

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

<value>=<NRf> specifies the horizontal position of the marker.

Returns

<NRf>

Examples

CALCulate:PULSe:OGRAm:MEASview1:MARKer1:FVTime:X 2E-6 places Marker 1 (M1) at 2 µs on the Frequency vs Time trace.

CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:FVTime:Y (Query Only)

Returns the marker amplitude of the selected marker in Frequency vs Time display of Pulse-Ogram.

Conditions

Measurement views: Pulse-Ogram

Group

Calculate commands

Syntax

CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:FVTime:Y?

Arguments

<x> = 0,1,2,3,4 is the marker. The specified marker must be activated using the CALCulate:MARKer:ADD command.

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Returns

<NRf> Marker amplitude of the selected marker.

Examples

CALCulate:PULSe:OGRAm:MEASview1:MARKer1:FVTime:Y? might return -34.28,indicating Marker 1 (M1) is at the amplitude of -34.28 dBm.

CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:TVTime:DELTa:X (Query Only)

Returns the delta marker time for the selected marker in the Time vs Time display of Pulse-Ogram.

Conditions

Measurement views: Pulse-Ogram

Group

Calculate commands

Syntax

CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:TVTime:DELTa:X ?

Arguments

<x> = 1,2,3,4 is the marker. The specified marker must be activated using the CALCulate:MARKer:ADD command.

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Returns

<NRf> Delta marker time for the selected marker

Examples

CALCulate:PULSe:OGRAm:MEASview1:MARKer1:TVTime:DELTa:X? might return -197.57E-6, indicating that the delta marker time for Marker 1 (M1) is 197.57 µs.

CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:TVTime:DELTa:Y (Query Only)

Queries the delta marker amplitude for the selected marker in the Time vs Time display of Pulse-Ogram.

Conditions

Measurement views: Pulse-Ogram

Group

Calculate commands

Syntax

`CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:TVTime:DELTa:Y ?`

Arguments

<x> = 1, 2, 3, 4 is the marker. The specified marker must be activated using the CALCulate:MARKer:ADD command.

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Returns

<NRf> Delta marker amplitude for the selected marker

Examples

`CALCulate:PULSe:OGRAm:MEASview1:MARKer1:TVTime:DELTa:Y?` might return `-813.10E-3`, indicating that the delta marker amplitude for Marker 1 (M1) is -813.10E-3.

CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:TVTime:MAXimum (No Query Form)

Moves the selected marker to the highest peak on the trace in Time vs Time display of Pulse-Ogram.

Conditions

Measurement views: Pulse-Ogram

Group

Calculate commands

Syntax

`CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:TVTime:MAXimum`

Arguments

<x> = 0,1,2,3,4 is the marker. The specified marker must be activated using the CALCulate:MARKer:ADD command.

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Returns

None

Examples

`CALCulate:PULSe:OGRAm:MEASview1:MARKer1:TVTime:MAXimum` moves Marker 1(M1) to the highest peak on the trace of PulseOgram Measurement View 1.

CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:TVTime:X

Sets or Queries the horizontal position of the selected marker in Time vs Time display of Pulse-Ogram.

Conditions

Measurement views: Pulse-Ogram

Group

Calculate commands

Syntax

```
CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:TVTime:X <value>
CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:TVTime:X ?
```

Arguments

<x> = 0,1,2,3,4 is the marker. The specified marker must be activated using the CALCulate:MARKer:ADD command.

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

`<value>=<NRf>` specifies the horizontal position of the marker

Returns

`<NRf>`

Examples

`CALCulate:PULSe:OGRAm:MEASview1:MARKer1:TVTime:X 2E-6` places Marker 1 (M1) at 2 µs on the Time vs Time trace of PulseOgram Measurement View 1.

CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:TVTime:Y (Query Only)

Returns the marker amplitude of the selected marker in Time vs Time display of Pulse-Ogram.

Conditions

Measurement views: Pulse-Ogram

Group

Calculate commands

Syntax

```
CALCulate:PULSe:OGRAm[:MEASview<y>]:MARKer<x>:TVTime:Y?
```

Arguments

<x> = 0,1,2,3,4 is the marker. The specified marker must be activated using the CALCulate:MARKer:ADD command.

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Returns

`<NRf>` Marker amplitude of the selected marker

Examples

`CALCulate:PULSe:OGRAm:MEASview1:MARKer1:TVTime:Y?` might return `-44.85`, indicating Marker 1 (M1) is at the amplitude of -44.85 dBm.

CALCulate:PULSe:STATistics[:MEASview<y>]:FFT:INDicator:X

Sets or queries the current bin number where the indicator is positioned from the histogram display.Sets or queries the frequency in Hz for the X indicator. This command is valid when `DISPlay:PULSe:STATistics:PLOT` is set to FFT.

Conditions

Measurement views: Pulse statistics

Group

Calculate commands

Syntax

```
CALCulate:PULSe:STATistics[:MEASview<y>]:FFT:INDicator:X <value>
CALCulate:PULSe:STATistics[:MEASview<y>]:FFT:INDicator:X?
```

Related Commands

CALCulate:PULSe:STATistics[:MEASview<y>]:FFT:INDicator:Y? (on page 543)

Arguments

`<value>::=<NRf>` specifies the indicator X value in Hz.

Examples

`CALCulate:PULSe:STATistics:MEASview2:FFT:INDicator:X 1000` sets the FFT indicator X value to 1 kHz.

CALCulate:PULSe:STATistics[:MEASview<y>]:FFT:INDicator:Y? (Query)Only

Returns the frequency level in dB for the Y indicator. This command is valid when `DISPlay:PULSe:STATistics:PLOT` is set to FFT.

Conditions

Measurement views: Pulse statistics

Group

Calculate commands

Syntax

```
CALCulate:PULSe:STATistics[:MEASview<y>]:FFT:INDicator:Y?
```

Related Commands

CALCulate:PULSe:STATistics[:MEASview<y>]:FFT:INDicator:X (on page 543)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<NRf>` the indicator Y value in dB.

Examples

CALCulate:PULSe:STATistics:MEASview2:FFT:INDicator:Y? might return -34.5 indicating that the FFT indicator is at -34.5 dB.

CALCulate:PULSe:STATistics[:MEASview<y>]:HISTogram:INDicator:X

Sets or queries the histogram indicator X value. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to histogram.

Conditions

Measurement views: Pulse statistics

Group

Calculate commands

Syntax

```
CALCulate:PULSe:STATistics[:MEASview<y>]:HISTogram:INDicator:X <value>
CALCulate:PULSe:STATistics[:MEASview<y>]:HISTogram:INDicator:X?
```

Related Commands

CALCulate:PULSe:STATistics[:MEASview<y>]:HISTogram:INDicator:Y? (on page 544)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<value>::=<NRf> specifies the indicator X value.

Examples

CALCulate:PULSe:STATistics:MEASview2:HISTogram:INDicator:X 0.3 sets the histogram indicator X value to 0.3.

CALCulate:PULSe:STATistics[:MEASview<y>]:HISTogram:INDicator:Y? (Query Only)

Queries the Histogram indicator Y value. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to Histogram.

Conditions

Measurement views: Pulse statistics

Group

Calculate commands

Syntax

```
CALCulate:PULSe:STATistics[:MEASview<y>]:HISTogram:INDicator:Y?
```

Related Commands

CALCulate:PULSe:STATistics[:MEASview<y>]:HISTogram:INDicator:X (on page 544)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf> indicates the indicator Y value.

Examples

CALCulate:PULSe:STATistics:MEASview2:HISTogram:INDicator:Y? might return 20 to indicate 20 counts in the histogram bin in dB.

CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:DELTa:X?

Returns the delta marker time for the selected marker on the pulse trace. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to Time Trend.

Conditions

Measurement views: Pulse statistics with specified Measurement view number

Group

Calculate commands

Syntax

CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:DELTa:X?

Related Commands

DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT (on page 940)

CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:DELTa:Y? (on page 545)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

<x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Returns

<NRf> Delta marker time in seconds for the selected marker.

Examples

CALCulate:PULSe:STATistics:MEASview1:MARKer1:DELTa:X? might return 3.01e-3, indicating that the delta marker time is 3.01 ms on the trace of pulse statistics measurement view 1.

CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:DELTa:Y?

Returns the delta marker amplitude for the selected marker on the pulse trace. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to Time Trend.

Conditions

Measurement views: Pulse statistics with specified Measurement view number

Group

Calculate commands

Syntax

```
CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:DELTa:Y?
```

Related Commands

DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT (on page 940)

CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:DELTa:X? (on page 545)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

<x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Returns

<NRf> Delta marker amplitude for the selected marker, in the unit of the current measurement.

Examples

`CALCulate:PULSe:STATistics:MEASview1:MARKer1:DELTa:Y?` might return 2.345, indicating that the delta marker amplitude is 2.345 dB on the trace of pulse statistics measurement view 1.

CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:MAXimum

Moves the selected marker to the highest peak on the pulse statistics trace. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to Time Trend.

Conditions

Measurement views: Pulse statistics with specified Measurement view number

Group

Calculate commands

Syntax

```
CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:MAXimum
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

<x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Examples

`CALCulate:PULSe:STATistics:MEASview1:MARKer1:MAXimum` moves Marker 1 (M1) to the highest peak on the trace of pulse statistics measurement view 1.

CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:PEAK:HIGHer

Moves the selected marker to the next peak higher in amplitude on the pulse statistics trace. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to Time Trend.

Conditions

Measurement views: Pulse statistics with specified Measurement view number

Group

Calculate commands

Syntax

`CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:PEAK:HIGHer`

Related Commands

CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:PEAK:LOWer (on page 548)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

<x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Examples

`CALCulate:PULSe:STATistics:MEASview1:MARKer1:PEAK:HIGHer` moves Marker 1 (M1) to the next peak higher in amplitude on the trace of pulse statistics measurement view 1.

CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:PEAK:LEFT

Moves the selected marker to the next peak to the left on the pulse statistics trace. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to Time Trend.

Conditions

Measurement views: Pulse statistics with specified Measurement view number

Group

Calculate commands

Syntax

`CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:PEAK:LEFT`

Related Commands

CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:PEAK:RIGHT (on page 548)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

<x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Examples

`CALCulate:PULSe:STATistics:MEASview1:MARKer1:PEAK:LEFT` moves Marker 1 (M1) to the next peak to the left on the trace of pulse statistics measurement view 1.

CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:PEAK:LOWer

Moves the selected marker to the next peak lower in amplitude on the pulse statistics trace. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to Time Trend.

Conditions

Measurement views: Pulse statistics with specified Measurement view number

Group

Calculate commands

Syntax

`CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:PEAK:LOWer`

Related Commands

CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (on page 546)

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

`<x>` = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Examples

`CALCulate:PULSe:STATistics:MEASview1:MARKer1:PEAK:LOWer` moves Marker 1 (M1) to the next peak lower in amplitude on the trace of pulse statistics measurement view 1.

CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:PEAK:RIGHt

Moves the selected marker to the next peak to the right on the pulse statistics trace. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to Time Trend.

Conditions

Measurement views: Pulse statistics with specified Measurement view number

Group

Calculate commands

Syntax

`CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:PEAK:RIGHt`

Related Commands

CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:PEAK:LEFT (on page 547)

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

`<x>` = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Examples

`CALCulate:PULSe:STATistics[:MEASview1:MARKer1:PEAK:RIGHt` moves Marker 1 (M1) to the next peak to the right on the trace of pulse statistics measurement view 1.

CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:X

Sets or queries the horizontal position of the selected marker in the pulse statistics view. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to Time Trend.

Conditions

Measurement views: Pulse statistics with specified Measurement view number

Group

Calculate commands

Syntax

```
CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:X <value>
CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:X?
```

Related Commands

CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:Y (on page 549)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

<x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Returns

`<NRf>` specifies the horizontal position of the marker.

Examples

`CALCulate:PULSe:STATistics:MEASvie1:MARKer1:X 3.01e-3` places Marker 1 (M1) at 3.01 ms on the trace of Pulse Statistics Measurement View 1.

CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:Y?

Queries the marker amplitude of the selected marker in the pulse statistics view. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to Time Trend.

Conditions

Measurement views: Pulse statistics with specified Measurement view number

Group

Calculate commands

Syntax

```
CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:Y?
```

Related Commands

CALCulate:PULSe:STATistics[:MEASview<y>]:MARKer<x>:X (on page 549)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

<x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Returns

`<NRf>` Amplitude of the specified marker in the unit of the current measurement.

Examples

`CALCulate:PULSe:STATistics:MEASview1:MARKer1:Y?` might return -28.173, indicating Marker 1 (M1) is at -28.173 dB on the trace of pulse statistics measurement view 1.

CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:DELTa:X

Returns the delta marker time for the selected marker on the pulse trace.

Conditions

Measurement views: Pulse trace with specified Measurement view number

Group

Calculate commands

Syntax

`CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:DELTa:X?`

Related Commands

CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:DELTa:Y? (on page 550)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

<x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Returns

`<NRf>` Delta marker time for the selected marker.

Examples

`CALCulate:PULSe:TRACe:MEASview1:MARKer1:DELTa:X?` might return 38.0E-9, indicating that the delta marker time is 38.0 ns on the trace of pulse trace measurement view 1.

CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:DELTa:Y

Returns the delta marker amplitude for the selected marker on the pulse trace.

Conditions

Measurement views: Pulse trace with specified Measurement view number

Group

Calculate commands

Syntax

```
CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:DELTa:Y?
```

Related Commands

CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:DELTa:X? (on page 550)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

<x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Returns

<NRf> Delta marker amplitude for the selected marker.

Examples

CALCulate:PULSe:TRACe:MEASview1:MARKer1:DELTa:Y? might return 23.45,indicating that the delta marker amplitude is 23.45 dB on the trace of pulse trace measurement view 1.

CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:MAXimum

Moves the selected marker to the highest peak on the pulse trace.

Conditions

Measurement views: Pulse trace with specified Measurement view number

Group

Calculate commands

Syntax

```
CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:MAXimum
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

<x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Examples

CALCulate:PULSe:TRACe:MEASview1:MARKer1:MAXimum moves Marker 1 (M1) to the highest peak on the trace of pulse trace measurement view 1.

CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:PEAK:HIGHer

Moves the selected marker to the next peak higher in amplitude on the pulse trace.

Conditions

Measurement views: Pulse trace with specified Measurement view number

Group

Calculate commands

Syntax

```
CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:PEAK:HIGHer
```

Related Commands

CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:PEAK:LOWer (on page 552)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

<x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Examples

`CALCulate:PULSe:TRACe:MEASview1:MARKer1:PEAK:HIGHer` moves Marker 1 (M1) to the next peak higher in amplitude on the trace of pulse trace measurement view 1.

CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:PEAK:LEFT

Moves the selected marker to the next peak to the left on the pulse trace.

Conditions

Measurement views: Pulse trace with specified Measurement view number

Group

Calculate commands

Syntax

```
CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:PEAK:LEFT
```

Related Commands

CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (on page 553)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

<x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Examples

`CALCulate:PULSe:TRACe:MEASview1:MARKer1:PEAK:LEFT` moves Marker 1 (M1) to the next peak to the left on the trace of pulse trace measurement view 1.

CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:PEAK:LOWer

Moves the selected marker to the next peak lower in amplitude on the pulse trace.

Conditions

Measurement views: Pulse trace with specified Measurement view number

Group

Calculate commands

Syntax

```
CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:PEAK:LOWer
```

Related Commands

CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (on page 551)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

<x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Examples

`CALCulate:PULSe:TRACe:MEASview1:MARKER1:PEAK:LOWer` moves Marker 1 (M1) to the next peak lower in amplitude on the trace of pulse trace measurement view 1.

CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:PEAK:RIGHt

Moves the selected marker to the next peak to the right on the pulse trace.

Conditions

Measurement views: Pulse trace with specified Measurement view number

Group

Calculate commands

Syntax

```
CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:PEAK:RIGHt
```

Related Commands

CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:PEAK:LEFT (on page 552)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

<x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Examples

`CALCulate:PULSe:TRACe:MEASview1:MARKer1:PEAK:RIGHt` moves Marker 1 (M1) to the next peak to the left on the trace of pulse trace measurement view 1.

CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:X

Sets or queries the horizontal position of the selected marker in the pulse trace view.

Conditions

Measurement views: Pulse trace with specified Measurement view number

Group

Calculate commands

Syntax

```
CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:X <value>
CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:X?
```

Related Commands

CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:Y? (on page 554)

Arguments

<value>::=<NRf> specifies the horizontal position of the marker.

NOTE: To set the horizontal position, the marker must be on the trace of the pulse selected using the DISPlay:PULSe:SELect:NUMBer command. You cannot put the marker out of the horizontal range of the pulse.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

<x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Returns

<NRf>

Examples

CALCulate:PULSe:TRACe:MEASview1:MARKer1:X 1.5 µs places Marker 1 (M1) at 1.5 µs on the trace.

CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:Y

Queries the marker amplitude of the selected marker in the pulse trace view.

Conditions

Measurement views: Pulse trace with specified Measurement view number

Group

Calculate commands

Syntax

```
CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:Y?
```

Related Commands

CALCulate:PULSe:TRACe[:MEASview<y>]:MARKer<x>:X? (on page 553)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

<x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Returns

<NRf> Marker amplitude of the selected marker.

Examples

CALCulate:PULSe:TRACe:MEASview1:MARKer1:Y? might return –28.86, indicating Marker 1 (M1) is at -28.86 dBm on the trace of pulse trace measurement view 1.

CALCulate:SEARch:LIMit:FAIL? (Query Only)

Queries whether the waveform cuts across the limit in the search operation.

Conditions

Measurement views: All

Group

Calculate commands

Syntax

```
CALCulate:SEARch:LIMit:FAIL?
```

Arguments

None

Returns

{ 0 | 1 }

0 represents Pass, indicating that the waveform does not cut across the limit.

1 represents Fail, indicating that the waveform cuts across the limit.

Examples

`CALCULATE:SEARCH:LIMIT:FAIL?` might return 1, indicating that the waveform cuts across the limit (Fail).

CALCulate:SEARch:LIMit:MATCh:BEEP[:STATe]

Determines whether or not to beep on match during run or replay in the search operation.

Conditions

Measurement views: All

Group

Calculate commands

Syntax

```
CALCulate:SEARch:LIMit:MATCh:BEEP[:STATe] { OFF | ON | 0 | 1 }
CALCulate:SEARch:LIMit:MATCh:BEEP[:STATe]?
```

Arguments

`OFF` or 0 disables to beep on match.

`ON` or 1 enables to beep on match.

Examples

`CALCULATE:SEARCH:LIMIT:MATCH:BEEP:STATE ON` enables to beep on match.

CALCulate:SEARch:LIMit:MATCh:SACQuire[:STATe]

Determines whether or not to stop acquiring data on match during run or replay in the search operation.

Conditions

Measurement views: All

Group

Calculate commands

Syntax

```
CALCulate:SEARch:LIMit:MATCh:SACQuire[:STATe] { OFF | ON | 0 | 1 }
CALCulate:SEARch:LIMit:MATCh:SACQuire[:STATe]?
```

Arguments

`OFF` or 0 disables to stop acquiring data on match.

`ON` or 1 enables to stop acquiring data on match.

Examples

`CALCULATE:SEARCH:LIMIT:MATCH:SACQUIRE:STATE ON` enables to stop acquiring data on match.

CALCulate:SEARch:LIMit:MATCh:SDATa[:STATe]

Determines whether or not to save automatically (AutoSave) acquisition data on match during run in the search operation.

Conditions

Measurement views: All

Group

Calculate commands

Syntax

```
CALCulate:SEARch:LIMit:MATCh:SDATa[:STATe] { OFF | ON | 0 | 1 }
CALCulate:SEARch:LIMit:MATCh:SDATa[:STATe]?
```

Arguments

`OFF` or 0 disables the AutoSave.

`ON` or 1 enables to save acquisition data automatically on match.

The data is saved to a file with the name:

Set NAME variable*-yyyy.mm.dd.hh.mm.ss.sss.tiq

Where

Set NAME variable* is the file name that was last specified.

yyyy.mm.dd and hh.mm.ss.sss represent date and 24 hour time.

The file extension is .tiq.

Example: SAVED-2007.03.20.12.34.567.tiq

For the directory of file, refer to Specifying the File (on page 237) in the MMEMory command section.

Examples

`CALCULATE:SEARCH:LIMIT:MATCH:SDATA:STATE ON` enables to save acquisition data automatically on match.

CALCulate:SEARch:LIMit:MATCh:SPICture[:STATe]

Determines whether or not to save automatically (AutoSave) the whole screen

on match during run in the search operation.

Conditions

Measurement views: All

Group

Calculate commands

Syntax

```
CALCulate:SEARch:LIMit:MATCh:SPICture[:STATe] { OFF | ON | 0 | 1 }
CALCulate:SEARch:LIMit:MATCh:SPICture[:STATe]?
```

Arguments

`OFF` or 0 disables the AutoSave.

`ON` or 1 enables to save the whole screen automatically on match.

The picture is saved to a file with the name:

Set NAME variable*-yyyy.mm.dd.hh.mm.ss.sss.png

Where

Set NAME variable* is the file name that was last specified.

yyyy.mm.dd and hh.mm.ss.sss represent date and 24 hour time.

The file extension is .png.

Example: SAVED-2007.03.20.12.34.567.png

For the directory of file, refer to Specifying the File (on page 237) in the MMEMory command section.

Examples

`CALCULATE:SEARCH:LIMIT:MATCH:SPICTURE:STATE ON` enables to save the whole screen automatically on match.

CALCulate:SEARch:LIMit:MATCh:STRace[:STATe]

Determines whether or not to save automatically (AutoSave) the spectrum trace on match during run in the search operation.

Conditions

Measurement views: All

Group

Calculate commands

Syntax

```
CALCulate:SEARch:LIMit:MATCh:STRace[:STATe] { OFF | ON | 0 | 1 }
CALCulate:SEARch:LIMit:MATCh:STRace[:STATe]?
```

Arguments

`OFF` or 0 disables the AutoSave.

`ON` or 1 enables to save the spectrum trace automatically on match.

The trace is saved to a file with the name:

Set NAME variable*-yyyy.mm.dd.hh.mm.ss.sss.Specan

Where

Set NAME variable* is the file name that was last specified.

yyyy.mm.dd and hh.mm.ss.sss represent date and 24 hour time.

The file extension is .Specan.

Example: SAVED-2007.03.20.12.34.567.Specan

For the directory of file, refer to Specifying the File (on page 237) in the MMEMory command section.

Examples

`CALCULATE:SEARCH:LIMIT:MATCH:STRACE:STATE ON` enables to save the spectrum trace automatically on match.

CALCulate:SEARch:LIMit:OPERation

Sets or queries the search condition.

Conditions

Measurement views: All

Group

Calculate commands

Syntax

```
CALCulate:SEARch:LIMit:OPERation { LT | GT | IMASk | OMASk }
CALCulate:SEARch:LIMit:OPERation?
```

Related Commands

CALCulate:[:SEARch|:TEST]:LIMit[:STATe]:MATCh:FILE:LOCation (on page 343)

Arguments

The following table lists the arguments.

Argument	Meaning
LT	The data is less than the limit value.
GT	The data is greater than the limit value.
IMASk	The data is inside the limit mask.
OMASk	The data is outside the limit mask.

You can select the data using the CALCulate:SEARch:LIMit:OPERation:FEED command.

You can set the limit value using the `CALCulate:[:SEARch|:TEST]:LIMit[:STATe]:MATCh:FILE:LOCation` command.

You can store and load the limit mask using the `CALCulate:SEARch:LIMit:OPERation:MASK:STORE` and `CALCulate:SEARch:LIMit:OPERation:MASK:LOAD` commands.

Examples

`CALCULATE:SEARCH:LIMIT:OPERATION GT` selects "the data is greater than the limit value" for the search condition.

CALCulate:SEARch:LIMit:OPERation:FEED

Sets or queries the data flow to be fed in the search operation.

Conditions

Measurement views: All

Group

Calculate commands

Syntax

```
CALCulate:SEARch:LIMit:OPERation:FEED <view>,<trace>
CALCulate:SEARch:LIMit:OPERation:FEED?
```

Arguments

`<view>::=<string>` and `<trace>::=<string>` are listed in the following table.

Source data of the search operation

<view>	<trace>	Meaning
"Spectrum"	"Trace 1" [1]	Trace 1 in the Spectrum view.
	"Trace 2" [1]	Trace 2 in the Spectrum view.
	"Trace 3" [1]	Trace 3 in the Spectrum view.
	"Math Trace"	Math trace in the Spectrum view.
	"Spectrogram Trace"	Spectrogram trace in the Spectrum view.

[1] There is a space character between Trace and the number.

Examples

`CALCULATE:SEARCH:LIMIT:OPERATION:FEED "Spectrum","Trace 1"` selects the Trace 1 in the Spectrum measurement view for the search operation.

`CALCULATE:SEARCH:LIMIT:OPERATION:FEED?` might return `"Spectrum","Math Trace"`, indicating that the math trace is used as the source data in the search operation.

CALCulate:SEARch:LIMit:OPERation:MASK:LOAD (No Query Form)

Loads the specified limit mask file for the search operation.

Conditions

Measurement views: All

Group

Calculate commands

Syntax

```
CALCulate:SEARch:LIMit:OPERation:MASK:LOAD <file_name>
```

Arguments

<file_name>::=<string> specifies the file to load the limit mask from.The file extension is .lmt. You can omit the extension.

For the directory of file, refer to Specifying the File (on page 237) in the MMEMory command section.

Examples

CALCULATE:SEARCH:LIMIT:OPERATION:MASK:LOAD"Limit1" loads the limit mask from the Limit1.lmt file.

CALCulate:SEARch:LIMit:OPERation:MASK:STORE (No Query Form)

Stores the limit mask to a specified file in the search operation.

Conditions

Measurement views: All

Group

Calculate commands

Syntax

```
CALCulate:SEARch:LIMit:OPERation:MASK:STORE <file_name>
```

Arguments

<file_name>::=<string> specifies the file to store the limit mask to.The file extension is .lmt. You can omit the extension.

For the directory of file, refer to Specifying the File (on page 237) in the MMEMory command section.

Examples

CALCULATE:SEARCH:LIMIT:OPERATION:MASK:STORE "Limit1" stores the limit mask to the Limit1.lmt file.

CALCulate:SEARch:LIMit:OPERation:SLIMit

Sets or queries the limit value (in dB) in the search operation.

Conditions

Measurement views: All

Group

Calculate commands

Syntax

```
CALCulate:SEARch:LIMit:OPERation:SLIMit <value>
CALCulate:SEARch:LIMit:OPERation:SLIMit?
```

Related Commands

CALCulate:SEARch:LIMit:OPERation (on page 558)

Arguments

<value>::=<NRf> specifies the limit value in the search operation.Range: -100 to +100 dBm.

Examples

CALCULATE:SEARCH:LIMIT:OPERATION:SLIMIT -20 sets the limit value to -20 dBm.

CALCulate:SEARch:LIMit:REPort:DATA? (Query Only)

Returns the frequency range(s) that satisfy the search condition.

Conditions

Measurement views: All

Group

Calculate commands

Syntax

`CALCulate:SEARch:LIMit:REPort:DATA?`

Arguments

None

Returns

`<num_range>,<range(1)>,<range(2)>,...,<range(n)>`

Where

`<num_range>::=<NR1>` is the number of ranges that satisfy the condition.

`<range(n)>::="<lower_freq(n)>, <upper_freq(n)>"` (string) represents the nth frequency range that satisfy the search condition in ascending order.

`<lower_freq(n)>` and `<upper_freq(n)>` are the lower and upper frequencies of the range #n, respectively.

Examples

`CALCULATE:SEARCH:LIMIT:REPORT:DATA?` might return 2,"1.4800E+9,1.5001E+9","1.5002E+9, 1.5200E+9", indicating that the search condition is satisfied in these two ranges 1.48 to 1.5001 GHz and 1.5002 to 1.52 GHz.

CALCulate:SEARch:LIMit:REPort:POINts? (Query Only)

Returns the number of frequency range(s) that satisfy the search condition.

Conditions

Measurement views: All

Group

Calculate commands

Syntax

`CALCulate:SEARch:LIMit:REPort:POINts?`

Arguments

None

Returns

`<number>::=<NR1>` represents the number of frequency range(s) that satisfy the search condition.

Examples

`CALCULATE:SEARCH:LIMIT:REPORT:POINTS?` might return 5, indicating that five ranges satisfy the search condition.

CALCulate:SEARch:LIMit:STATe

Sets or queries the search function (enabled or disabled).

Conditions

Measurement views: All

Group

Calculate commands

Syntax

```
CALCulate:SEARch:LIMit:STATe { OFF | ON | 0 | 1 }
CALCulate:SEARch:LIMit:STATe?
```

Arguments

`OFF` or 0 disables the search function.

`ON` or 1 enables the search function.

Examples

`CALCULATE:SEARCH:LIMIT:STATE ON` enables the search function.

CALCulate:SEARch:LIMit:STATe:MATCh:ACQuisition:FORMat

Sets and queries the format in which the acquisition data to be saved in mask test control settings.

Conditions

Measurement view: Spectrum

Group

Calculate commands

Syntax

```
CALCulate:SEARch:LIMit:STATe:MATCh:ACQuisition:FORMat {TIQ | CSV | MAT}
CALCulate:SEARch:LIMit:STATe:MATCh:ACQuisition:FORMat?
```

Arguments

`TIQ` saves acquisition data in `.Tiq` file format.

`CSV` saves acquisition data in `.csv` file format.

`MAT` saves acquisition data in `.Mat` file format.

Returns

`<String>` Either TIQ , CSV or MAT

Examples

`CALCULATE:SEARCH:LIMIT:STATE:MATCH:ACQUISITION:FORMAT` might return CSV, indicating that the acquisition data is saved in .csv file format.

CALCulate:SEARch:LIMit:STATe:MATCh:BEEP:STATe

Determines the enabling and disabling of Beep check box in mask test control settings.

Conditions

Measurement view: Spectrum

Group

Calculate commands

Syntax

```
CALCulate:SEARch:LIMit:STATe:MATCh:BEEP:STATe {0 | OFF | 1 | ON}
CALCulate:SEARch:LIMit:STATe:MATCh:BEEP:STATe?
```

Arguments

`0` or OFF disables the Beep check box

`1` or ON enables the Beep check box

Returns

<Boolean> 0,1

Examples

`CALCULATE:SEARCH:LIMIT:STATE:MATCH:BEEP:STATE` might return 0, indicating that the Beep check box is disabled.

CALCulate:SEARch:LIMit:STATe:MATCh:COUNt:STATe

Determines the enabling and disabling of Max files check box in mask test control settings.

Conditions

Measurement view: Spectrum

Group

Calculate commands

Syntax

```
CALCulate:SEARch:LIMit:STATe:MATCh:COUNt:STATe {0 | OFF | 1 | ON}
CALCulate:SEARch:LIMit:STATe:MATCh:COUNt:STATe?
```

Arguments

`0` or `OFF` disables the Max files check box

`1` or `ON` enables the Max files check box

Returns

`<Boolean>` 0,1

Examples

`CALCULATE:SEARCH:LIMIT:STATE:MATCH:COUNT:STATE` might return 0 indicating that the Max files check box is disabled.

CALCulate:SEARch:LIMit:STATe:MATCh:COUNt:STATe:MAXimum

Sets or queries the Max files value in mask test control settings.

Conditions

Measurement view: Spectrum

Group

Calculate commands

Syntax

```
CALCulate:SEARch:LIMit:STATe:MATCh:COUNt:STATe:MAXimum<value>
CALCulate:SEARch:LIMit:STATe:MATCh:COUNt:STATe:MAXimum?
```

Arguments

`<value> = <NR1>` Range is 1 to 10000

Returns

`<NR1>`

Examples

`CALCULATE:SEARCH:LIMIT:STATE:MATCH:COUNT:STATE:MAXIMUM` might return 250 indicating the value of max files given is 250.

CALCulate:SEARch:LIMit:STATe:MATCh:FILE:LOCation

Sets or queries the location of the file in mask test control settings.

Conditions

Measurement view: Spectrum

Group

Calculate commands

Syntax

```
CALCulate:SEARch:LIMit:STATe:MATCh:FILE:LOCation <file_name>
CALCulate:SEARch:LIMit:STATe:MATCh:FILE:LOCation?
```

Arguments

`<file_name> = <String>` Specifies the location of the file

Returns

`<String>`

Examples

`CALCULATE:SEARCH:LIMIT:STATE:MATCH:FILE:LOCATION` might return `"C:\SignalVu-PC Files"`, indicating the location of the file is `"C:\SignalVu-PC Files"`.

CALCulate:SEARch:LIMit:STATe:MATCh:FILE:NAME:BASE

Sets and queries the file name in mask test control settings.

Conditions

Measurement view: Spectrum

Group

Calculate commands

Syntax

```
CALCulate:SEARch:LIMit:STATe:MATCh:FILE:NAME:BASE <file_name>
CALCulate:SEARch:LIMit:STATe:MATCh:FILE:NAME:BASE?
```

Arguments

`<file_name> = <String>` Specifies the name of the file

Returns

`<String>`

Examples

`CALCULATE:SEARCH:LIMIT:STATE:MATCH:FILE:NAME:BASE` might return `"Sample1"`, indicating the name of the file is `"Sample1"`.

CALCulate:SEARch:LIMit:STATe:MATCh:PICTure:FORMat

Sets and queries the format in which the picture to be saved in mask test control settings.

Conditions

Measurement view: Spectrum

Group

Calculate commands

Syntax

```
CALCulate:SEARch:LIMit:STATe:MATCh:PICTure:FORMat {PNG | JPG | BMP}
CALCulate:SEARch:LIMit:STATe:MATCh:PICTure:FORMat?
```

Arguments

`PNG` saves picture in png format.

`JPG` saves picture in jpg format.

`BMP` saves picture in bmp format.

Returns

`<String>` Either PNG, JPG or BMP

Examples

`CALCULATE:SEARCH:LIMIT:STATE:MATCH:PICTURE:FORMAT` might return `JPG`, indicating that the Picture is saved in jpg format.

CALCulate:SEARch:LIMit:STATe:MATCh:SACQuire:STATe

Determines the enabling and disabling of Stop check box in mask test control settings.

Conditions

Measurement view: Spectrum

Group

Calculate commands

Syntax

```
CALCulate:SEARch:LIMit:STATe:MATCh:SACQuire:STATe {0 | OFF | 1 | ON}
CALCulate:SEARch:LIMit:STATe:MATCh:SACQuire:STATe?
```

Arguments

0 or OFF disables the stop check box

1 or ON enables the stop check box

Returns

<Boolean> 0,1

Examples

CALCULATE:SEARCH:LIMIT:STATE:MATCH:SACQUIRE:STATE might return 1,indicating that the stop check box is enabled.

CALCulate:SEARch:LIMit:STATe:MATCh:SDATa:STATe

Determines the enabling and disabling of Save acquisition data check box in mask test control settings.

Conditions

Measurement view: Spectrum

Group

Calculate commands

Syntax

```
CALCulate:SEARch:LIMit:STATe:MATCh:SDATa:STATe {0 | OFF | 1 | ON}
CALCulate:SEARch:LIMit:STATe:MATCh:SDATa:STATe?
```

Arguments

0 or OFF disables the Save acquisition data check box

1 or ON enables the Save acquisition data check box

Returns

<Boolean> 0,1

Examples

CALCULATE:SEARCH:LIMIT:STATE:MATCH:SDATA:STATE might return 1, indicating that the Save acquisition data check box is enabled.

CALCulate:SEARch:LIMit:STATe:MATCh:SPICture:STATe

Determines the enabling and disabling of Save picture check box in mask test control settings.

Conditions

Measurement view: Spectrum

Group

Calculate commands

Syntax

```
CALCulate:SEARch:LIMit:STATe:MATCh:SPICture:STATe {0 | OFF | 1 | ON}
CALCulate:SEARch:LIMit:STATe:MATCh:SPICture:STATe?
```

Arguments

0 or OFF disables the Save picture check box

1 or ON enables the Save picture check box

Returns

<Boolean> 0,1

Examples

CALCULATE:SEARCH:LIMIT:STATE:MATCH:SPICTURE:STATE might return 1, indicating that the Save picture check box is enabled.

CALCulate:SEARch:LIMit:STATe:MATCh:STRace:STATe

Determines the enabling and disabling of Save trace check box in mask test control settings.

Conditions

Measurement view: Spectrum

Group

Calculate commands

Syntax

```
CALCulate:SEARch:LIMit:STATe:MATCh:STRace:STATe {0 | OFF | 1 | ON}
CALCulate:SEARch:LIMit:STATe:MATCh:STRace:STATe?
```

Arguments

0 or OFF disables the Save trace check box

1 or ON enables the Save trace check box

Returns

<Boolean> 0,1

Examples

CALCULATE:SEARCH:LIMIT:STATE:MATCH:STRACE:STATE might return 0, indicating that the Save trace check box is disabled.

CALCulate:SEARch:LIMit:STATe:OPERation

Sets or queries the region in which the test should be performed in mask test xcontrol settings.

Conditions

Measurement view: Spectrum

Group

Calculate commands

Syntax

```
CALCulate:SEARch:LIMit:STATe:OPERation {GT | LT | OMAS | IMAS}
CALCulate:SEARch:LIMit:STATe:OPERation ?
```

Arguments

`GT` selects “Is greater than” as the region for the test to be performed

`LT` selects “Is less than” as the region for the test to be performed

`OMAS` selects “Is Outside mask” as the region for the test to be performed

`IMAS` selects “Is inside mask” as the region for the test to be performed

Returns

`<String>` Either GT , LT , OMAS or IMAS

Examples

`CALCULATE:SEARCH:LIMIT:STATE:OPERATION` might return LT, indicating “Is less than” is selected as the region for the test to be performed.

CALCulate:SEARch:LIMit:STATe:OPERation:FEED

Sets and queries the trace to which the test should be performed in mask test control settings.

Conditions

Measurement view: Spectrum

Group

Calculate commands

Syntax

```
CALCulate:SEARch:LIMit:STATe:OPERation:FEED <Arg>
CALCulate:SEARch:LIMit:STATe:OPERation:FEED?
```

Arguments

`<Arg>` =

`"Spectrum","Trace 1"` selects Trace 1 in spectrum

`"Spectrum","Trace 2"` selects Trace 2 in spectrum

`"Spectrum","Trace 3"` selects Trace 3 in spectrum

`"Spectrum","Math Trace"` selects math trace in spectrum

`"Spectrum","Spectrogram Trace"` selects Spectrogram trace in spectrum

Returns

`<String>` See Arguments

Examples

CALCULATE:SEARCH:LIMIT:STATE:OPERATION:FEED might return "Spectrum","Trace 1", indicating that the test should be performed on "Spectrum","Trace 1".

CALCulate:SEARch:LIMit:STATe:OPERation:MASK:LOAD (No Query Form)

Sets the location of the file which has to be loaded in mask test control settings.

Conditions

Measurement view: Spectrum

Group

Calculate commands

Syntax

```
CALCulate:SEARch:LIMit:STATe:OPERation:MASK:LOAD <file_name>
```

Arguments

<file_name> = <String> Specifies the location of the file

Returns

None

Examples

CALCULATE:SEARCH:LIMIT:STATE:OPERATION:MASK:LOAD "C:\SignalVu-PC Files" loads the data from the file "C:\SignalVu-PC Files".

CALCulate:SEARch:LIMit:STATe:OPERation:MASK:LOWer:STATe

Determines the enabling and disabling of Lower mask check box in mask test editor tab of mask test control settings.

Conditions

Measurement view: Spectrum

Group

Calculate commands

Syntax

```
CALCulate:SEARch:LIMit:STATe:OPERation:MASK:LOWer:STATe {0 | OFF | 1 | ON}
CALCulate:SEARch:LIMit:STATe:OPERation:MASK:LOWer:STATe?
```

Arguments

0 or OFF disables the Lower mask check box

1 or ON enables the Lower mask check box

Returns

<Boolean> 0,1

Examples

CALCULATE:SEARCH:LIMIT:STATE:OPERATION:MASK:LOWER:STATE might return 0, indicating that the Lower mask check box is disabled.

CALCulate:SEARch:LIMit:STATe:OPERation:MASK:NEW:AUTO (No Query Form)

Draws a new frequency mask automatically based on a reference trace.

Conditions

Measurement view: Spectrum, DPX, Spectrum

Group

Calculate commands

Syntax

CALCulate:SEARch:LIMit:STATe:OPERation:MASK:NEW:AUTO

Arguments

<x_margin>::<NRf> specifies the X margin (horizontal offset from the reference trace) in Hz.

<y_margin>::<NRf> specifies the Y margin (vertical offset from the reference trace) in dB.

<meas_ID>::<string> specifies the measurement view.

<trace_ID> specifies the reference trace.

The values for the above arguments are listed in the following table for each possible measurement view.

Measurement view	<meas_ID>	<trace_ID>
Spectrum	“specan”	TRACE1 (Trace 1)
		TRACE2 (Trace 2)
		TRACE3 (Trace 3)
		TRACE4 (Math trace
		TRACE5 (Spectrogram trace)
DPX, Spectrum	DPX	TRACE1 (+Peak trace)
		TRACE2 (–Peak trace)
		TRACE3 (Avg trace)
		TRACE4 (Math trace)

Examples

CALCULATE:SEARCH:LIMIT:STATE:OPERATION:MASK:NEW:AUTO "specan",TRACE1 ,7E+6,5 automatically draws a new frequency mask in the Spectrum view based on Trace 1 with a horizontal margin of 7 MHz and a vertical margin of 5 dB.

CALCulate:SEARch:LIMit:STATe:OPERation:MASK:STORE (No Query Form)

Sets the location of the file to which data has to be saved in mask test control settings.

Conditions

Measurement view: Spectrum

Group

Calculate commands

Syntax

```
CALCulate:SEARch:LIMit:STATe:OPERation:MASK:STORE <file_name>
```

Arguments

`<file_name> = <String>` Specifies the location of the file

Returns

None

Examples

`CALCULATE:SEARCH:LIMIT:STATE:OPERATION:MASK:STORE "C:\SignalVu-PC Files"` saves the data to the file `"C:\SignalVu-PC Files"`.

CALCulate:SEARch:LIMit:STATe:OPERation:MASK:UPPer:STATe

Determines the enabling and disabling of Upper mask check box in mask test editor tab of mask test control settings.

Conditions

Measurement view: Spectrum

Group

Calculate commands

Syntax

```
CALCulate:SEARch:LIMit:STATe:OPERation:MASK:UPPer:STATe {0 | OFF | 1 | ON}
CALCulate:SEARch:LIMit:STATe:OPERation:MASK:UPPer:STATe?
```

Arguments

`0` or `OFF` disables the Upper mask check box

`1` or `ON` enables the Upper mask check box

Returns

`<Boolean>` 0,1

Examples

`CALCULATE:SEARCH:LIMIT:STATE:OPERATION:MASK:UPPER:STATE` might return 0, indicating that the Upper mask check box is disabled.

CALCulate:SEARch:LIMit:STATe

Determines the enabling and disabling of Enable test check box in mask test

control settings.

Conditions

Measurement view: Spectrum

Group

Calculate commands

Syntax

```
CALCulate:SEARch:LIMit:STATe {0 | OFF | 1 | ON}
CALCulate:SEARch:LIMit:STATe?
```

Arguments

`0` or `OFF` disables the Enable test check box

`1` or `ON` enables the Enable test check box

Returns

<Boolean> 0,1

Examples

`CALCULATE:SEARCH:LIMIT:STATE` might return 1, indicating that the Enable test check box is enabled.

CALCulate:SEM:MARKer<x>:DELTa:X? (Query Only)

Returns the delta marker frequency for the selected marker on the Spectral Emissions Mask trace.

The parameter <x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: Spectral Emissions Mask

Group

Calculate commands

Syntax

```
CALCulate:SEM:MARKer<x>:DELTa:X?
```

Arguments

None

Returns

`<NRf>` Delta marker frequency for the selected marker.

Examples

`CALCulate:SEM:MARKer1:DELTa:X?` might return 1.28E+6, indicating that the delta marker frequency is 1.28 MHz.

CALCulate:SEM:MARKer<x>:DELTa:Y? (Query Only)

Returns the delta marker amplitude for the selected marker on the Spectral Emissions Mask trace.

The parameter <x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: Spectral Emissions Mask

Group

Calculate commands

Syntax

`CALCulate:SEM:MARKer<x>:DELTa:Y?`

Arguments

None

Returns

`<NRf>` Delta marker amplitude for the selected marker.

Examples

`CALCulate:SEM:MARKer1:DELTa:Y?` might return 23.45, indicating that the delta marker amplitude is 23.45 dB.

CALCulate:SEM:MARKer<x>:MAXimum (No Query Form)

Moves the selected marker to the highest peak on the Spectral Emissions Mask trace.

Conditions

Measurement views: Spectral Emissions Mask

Group

Calculate commands

Syntax

`CALCulate:SEM:MARKer<x>:MAXimum`

Arguments

None

Examples

`CALCulate:SEM:MARKer1:MAXimum` moves Marker 1 (M1) to the highest peak on the trace.

CALCulate:SEM:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the selected marker to the next peak higher in amplitude on the Spectral Emissions Mask trace.

Conditions

Measurement views: Spectral Emissions Mask

Group

Calculate commands

Syntax

`CALCulate:SEM:MARKer<x>:PEAK:HIGHer`

Arguments

None

Examples

`CALCulate:SEM:MARKer1:PEAK:HIGHer` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

CALCulate:SEM:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the selected marker to the next peak to the left on the Spectral Emissions Mask trace.

Conditions

Measurement views: Spectral Emissions Mask

Group

Calculate commands

Syntax

`CALCulate:SEM:MARKer<x>:PEAK:LEFT`

Arguments

None

Examples

`CALCulate:SEM:MARKer1:PEAK:LEFT` moves Marker 1 (M1) to the next peak to the left on the trace.

CALCulate:SEM:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the selected marker to the next peak lower in amplitude on the Spectral Emissions Mask trace.

Conditions

Measurement views: Spectral Emissions Mask

Group

Calculate commands

Syntax

`CALCulate:SEM:MARKer<x>:PEAK:LOWer`

Arguments

None

Examples

`CALCulate:SEM:MARKer1:PEAK:LOWer` moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

CALCulate:SEM:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the selected marker to the next peak to the right on the Spectral Emissions Mask trace.

Conditions

Measurement views: Spectral Emissions Mask

Group

Calculate commands

Syntax

`CALCulate:SEM:MARKer<x>:PEAK:RIGHt`

Arguments

None

Examples

`CALCulate:SEM:MARKer1:PEAK:RIGHt` moves Marker 1 (M1) to the next peak to the right on the trace.

CALCulate:SEM:MARKer<x>[:SET]:CENTer (No Query Form)

Moves the specified marker to the center frequency.

Conditions

Measurement views: Spectral Emissions Mask

Group

Calculate commands

Syntax

`CALCulate:SEM:MARKer<x>[:SET]:CENTer`

Arguments

None

Examples

`CALCulate:SEM:MARKer1[:SET]:CENTer` moves Marker 1 (M1) to the center frequency.

CALCulate:SEM:MARKer<x>:X

Sets or queries the horizontal position of the selected marker on the Spectral Emissions Mask trace.

Conditions

Measurement views: Spectral Emissions Mask

Group

Calculate commands

Syntax

`CALCulate:SEM:MARKer<x>:X <value>`

Arguments

<value>::=<NRf> specifies the horizontal position of the marker.

Examples

CALCULATE:SEM:MARKER1:X 800MHz places Marker 1 (M1) at 800 MHz on the spectrum trace.

CALCulate:SEM:MARKer<x>:Y? (Query Only)

Queries the marker amplitude of the selected marker on the Spectral Emissions Mask trace.

Conditions

Measurement views: Spectral Emissions Mask

Group

Calculate commands

Syntax

CALCulate:SEM:MARKer<x>:Y?

Arguments

None

Returns

<NRf> Marker amplitude of the selected marker.

Examples

CALCULATE:SEM:MARKER1:Y? might return -34.28, indicating Marker 1 (M1) is at -34.28 dBm.

CALCulate:SGRam[:MEASview<y>]:MARKer<x>:DELTa:X:FREQuency

Returns the delta marker frequency for the selected marker in the spectrogram.

Conditions

Measurement views: Spectrogram

Group

Calculate commands

Syntax

CALCulate:SGRam[:MEASview<y>]:MARKer<x>:DELTa:X:FREQuency?

Related Commands

CALCulate:SGRam[:MEASview<y>]:MARKer<x>:DELTa:Y? (on page 577)

Arguments

<x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Returns

<NRf> Delta marker frequency for the selected marker.

Examples

CALCulate:SGRam:MEASview1:MARKer1:DELTa:X:FREQuency? might return 5.95E+6,indicating that the delta marker frequency is 5.95 MHz.

CALCulate:SGRam[:MEASview<y>]:MARKer<x>:DELTa:X[:TIME] (Query Only)

Returns the delta marker time for the selected marker in the spectrogram.

Conditions

Measurement views: Spectrogram

Group

Calculate commands

Syntax

CALCulate:SGRam[:MEASview<y>]:MARKer<x>:DELTa:X[:TIME]?

Related Commands

CALCulate:SGRam[:MEASview<y>]:MARKer<x>:DELTa:Y? (on page 577)

Arguments

<x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Returns

<NRf> Delta marker time for the selected marker.

Examples

CALCulate:SGRam:MEASview1:MARKer1:DELTa:X:TIME? might return -1.84E-3,indicating that the delta marker time is -1.84 ms.

CALCulate:SGRam[:MEASview<y>]:MARKer<x>:DELTa:Y (Query Only)

Returns the delta marker amplitude for the selected marker in the spectrogram.

Conditions

Measurement views: Spectrogram

Group

Calculate commands

Syntax

CALCulate:SGRam[:MEASview<y>]:MARKer<x>:DELTa:Y?

Related Commands

CALCulate:SGRam[:MEASview<y>]:MARKer<x>:DELTa:X[:TIME] (on page 577)

Arguments

<x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Returns

<NRf> Delta marker amplitude for the selected marker.

Examples

CALCulate:SGRam:MEASview1:MARKer1:DELTa:Y? might return -8.45, indicating that the delta marker amplitude is -8.45 dB.

CALCulate:SGRam[:MEASview<y>]:MARKer<x>:MAXimum (No Query Form)

Moves the selected marker to the highest peak on a line in the spectrogram. The line is selected using the TRACe:SGRam:SELect:LINE command.

Conditions

Measurement views: Spectrogram

Group

Calculate commands

Syntax

CALCulate:SGRam[:MEASview<y>]:MARKer<x>:MAXimum

Arguments

<x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Examples

CALCulate:SGRam:MEASview1:MARKer1:MAXimum moves Marker 1 (M1) to the highest peak on the line.

CALCulate:SGRam[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the selected marker to the next peak higher in amplitude on a line in the spectrogram. The line is selected using the TRACe:SGRam:SELect:LINE command.

Conditions

Measurement views: Spectrogram

Group

Calculate commands

Syntax

```
CALCulate:SGRam[:MEASview<y>]:MARKer<x>:PEAK:HIGHer
```

Related Commands

CALCulate:SGRam[:MEASview<y>]:MARKer<x>:PEAK:LOWer (on page 579)

Arguments

<x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Examples

`CALCulate:SGRam:MEASview1:MARKer1:PEAK:HIGHer` moves Marker 1 (M1) to the next peak higher in amplitude on the line.

CALCulate:SGRam[:MEASview<y>]:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the selected marker to the next peak to the left on a line in the spectrogram.The line is selected using the TRACe:SGRam:SELect:LINE command.

Conditions

Measurement views: Spectrogram

Group

Calculate commands

Syntax

```
CALCulate:SGRam[:MEASview<y>]:MARKer<x>:PEAK:LEFT
```

Related Commands

CALCulate:SGRam[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (on page 580)

Arguments

<x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Examples

`CALCulate:SGRam:MEASview1:MARKer1:PEAK:LEFT` moves Marker 1 (M1) to the next peak to the left on the trace of Spectrum Measurement View 1.

CALCulate:SGRam[:MEASview<y>]:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the selected marker to the next peak lower in amplitude on a line in the spectrogram. The line is selected using the `TRACe:SGRam:SELect:LINE` command.

Conditions

Measurement views: Spectrogram

Group

Calculate commands

Syntax

`CALCulate:SGRam[:MEASview<y>]:MARKer<x>:PEAK:LOWer`

Related Commands

CALCulate:SGRam[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (on page 578)

Arguments

<x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Examples

`CALCulate:SGRam:MEASview1:MARKer1:PEAK:LOWer` moves Marker 1 (M1) to the next peak lower in amplitude on the line.

CALCulate:SGRam[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the selected marker to the next peak to the right on a line in the spectrogram.The line is selected using the TRACe:SGRam:SELect:LINE command.

Conditions

Measurement views: Spectrogram

Group

Calculate commands

Syntax

`CALCulate:SGRam[:MEASview<y>]:MARKer<x>:PEAK:RIGHt`

Related Commands

CALCulate:SGRam[:MEASview<y>]:MARKer<x>:PEAK:LEFT (on page 579)

Arguments

<x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Examples

`CALCulate:SGRam:MEASview1:MARKer1:PEAK:RIGHt` moves Marker 1 (M1) to the next peak to the right on the trace of Spectrogram Measurement View 1.

CALCulate:SGRam[:MEASview<y>]:MARKer<x>[:SET]:CENTer (No Query Form)

Sets the center frequency to the marker frequency in the spectrogram.

Conditions

Measurement views: Spectrogram

Group

Calculate commands

Syntax

```
CALCulate:SGRam[:MEASview<y>]:MARKer<x>[:SET]:CENTer
```

Arguments

<x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Examples

`CALCulate:SGRam:MEASview1:MARKer1:SET:CENTER` sets the center frequency to the Marker 1 frequency in the spectrogram.

CALCulate:SGRam[:MEASview<y>]:MARKer<x>:X:FREQuency

Sets or queries the marker frequency in the spectrogram.

Conditions

Measurement views: Spectrogram

Group

Calculate commands

Syntax

```
CALCulate:SGRam[:MEASview<y>]:MARKer<x>:X:FREQuency <value>
CALCulate:SGRam[:MEASview<y>]:MARKer<x>:X:FREQuency?
```

Related Commands

CALCulate:SGRam[:MEASview<y>]:MARKer<x>:Y (on page 582)

Arguments

<x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

`<value>::=<NRf>` specifies the marker frequency.Range: (center frequency) ± (span)/2.

Returns

<NRf> specifies the marker frequency.

Examples

`CALCulate:SGRam:MEASview1:MARKer1:X:FREQuency 800MHz` places Marker 1 (M1) at 800 MHz on the trace.

CALCulate:SGRam[:MEASview<y>]:MARKer<x>:X[:TIME]

Sets or queries the marker time in the spectrogram.

Conditions

Measurement views: Spectrogram

Group

Calculate commands

Syntax

```
CALCulate:SGRam[:MEASview<y>]:MARKer<x>:X[:TIME] <value>
CALCulate:SGRam[:MEASview<y>]:MARKer<x>:X[:TIME]?
```

Related Commands

CALCulate:SGRam[:MEASview<y>]:MARKer<x>:Y? (on page 582)

Arguments

<x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

`<value>::=<NRf>` specifies the marker time.

Returns

<NRf> specifies the marker time.

Examples

`CALCulate:SGRam:MEASview1:MARKer1:X:TIME -234.5us` places Marker 1 (M1) at -234.5 µs on the trace.

CALCulate:SGRam[:MEASview<y>]:MARKer<x>:Y (Query Only)

Queries the marker amplitude in the spectrogram.

Conditions

Measurement views: Spectrogram

Group

Calculate commands

Syntax

```
CALCulate:SGRam[:MEASview<y>]:MARKer<x>:Y?
```

Related Commands

CALCulate:SGRam[:MEASview<x>]:MARKer<y>:X:TIME (on page 582)

Arguments

<x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Returns

<NRf> Marker amplitude of the selected marker.

Examples

CALCulate:SGRam:MEASview1:MARKer1:Y? might return -34.28, indicating Marker 1 (M1) is at -34.28 dBm.

CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:DELTa:X? (Query Only)

Returns the delta marker frequency for the selected marker on the spectrum trace.

Conditions

Measurement views: Spectrum with the specified Measurement View Number.

The parameter <x> = 1 to 4.

Group

Calculate commands

Syntax

CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:DELTa:X? (QUERY)

Arguments

The parameter <x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

<NRf> Delta marker frequency for the selected marker.

Examples

CALCulate:SPECtrum:MEASview1:MARKer1:DELTa:X? might return 1.28E+6, indicating that the delta marker frequency is 1.28 MHz.

CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:DELTa:Y? (Query Only)

Returns the delta marker amplitude for the selected marker on the spectrum trace.

Conditions

Measurement views: Spectrum with the specified Measurement View Number

Group

Calculate commands

Syntax

CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:DELTa:Y? (QUERY)

Arguments

The parameter <x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

<NRf> Delta marker amplitude for the selected marker.

Examples

CALCulate:SPECtrum:MEASview1:MARKer1:DELTa:Y? might return 23.45, indicating that the delta marker amplitude is 23.45 dB.

CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:MAXimum

Moves the selected marker to the highest peak on the spectrum trace.

Conditions

Measurement view: Spectrum with the specified Measurement View Number

Group

Calculate commands

Syntax

CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:MAXimum

Arguments

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

<NRf>

Examples

CALCulate:SPECtrum:MEASview1:MARKer1:MAXimum moves Marker 1 (M1) to the highest peak on the trace of Spectrum Measurement View 1.

CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the selected marker to the next peak higher in amplitude on the spectrum trace.

Conditions

Measurement views: Spectrum with the specified Measurement View Number.

Group

Calculate commands

Syntax

`CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:PEAK:HIGHer`

Arguments

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

<NRf>

Examples

`CALCulate:SPECtrum:MEASview1:MARKer1:PEAK:HIGHer` moves Marker 1 (M1) to the next higher peak in amplitude on the trace of Spectrum Measurement View 1

CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the selected marker to the next peak to the left on the spectrum trace.

Conditions

Measurement views: Spectrum with the specified Measurement View Number.

Group

Calculate commands

Syntax

`CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:PEAK:LEFT`

Arguments

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

<NRf>

Examples

`CALCulate:SPECtrum:MEASview1:MARKer1:PEAK:LEFT` moves Marker 1 (M1) to the next peak to the right on the trace of Spectrum Measurement View 1.

CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the selected marker to the next peak lower in amplitude on the spectrum trace.

Conditions

Measurement views: Spectrum with the specified Measurement View Number.

Group

Calculate commands.

Syntax

`CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:PEAK:LOWer`

Arguments

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

<NRf>

Examples

`CALCulate:SPECtrum:MEASview1:MARKer1:PEAK:LOWer` moves Marker 1 (M1) to the next lower peak in amplitude on the trace of Spectrum Measurement View 1.

CALCulate:SPECtrum[:MEASview<y>]:MARKer<x:PEAK:RIGHt (No Query Form)

Moves the selected marker to the next peak to the right on the spectrum trace.

Conditions

Measurement views: Spectrum with the specified Measurement View Number

Group

Calculate commands

Syntax

`CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:PEAK:RIGHt`

Arguments

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

<NRf>

Examples

`CALCulate:SPECtrum:MEASview1:MARKer1:PEAK:RIGHt` moves Marker 1 (M1) to the next peak to the right on the trace of Spectrum Measurement View 1.

CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:POWer:DENSity? (Query Only)

Returns the power density of the specified marker in the Spectrum measurement.

Conditions

Measurement view: Spectrum with the specified Measurement View Number

The parameter <x> = 0 to 4.

Group

Calculate commands

Syntax

`CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:POWer:DENsity? (QUERY)`

Arguments

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

`<NR2>` = a floating point value without exponents. It is the power density of the selected marker in dBm/Hz.

Examples

`CALCulate:SPECtrum:MEASview1:MARKer1:POWer:DENsity?` might return `-80.2412563552`, indicating that the power density is -80.24 dBm/Hz.

CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:POWer:IDENSity?

Returns the integrated power density reading between the MR (reference marker) and the selected marker in the Spectrum measurement.

Conditions

Measurement view: Spectrum with the specified Measurement View Number.

The parameter <x> = 0 to 4.

Group

Calculate commands

Syntax

`CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:POWer:IDENSity? <QUERY>`

Arguments

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

`<NR2>` = a floating point value without exponents. It is the integrated power density reading between MR (reference marker) and the selected marker in dBm/Hz.

Examples

`CALCulate:SPECtrum:MEASview1:MARKer1:POWer:IDENSity?` might return `-78.5625446112`, indicating that the integrated power density is -78.56 dBm/Hz.

CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:POWer:INTEgrated? (Query Only)

Returns the integrated power reading between the MR (reference marker) and the selected marker in the Spectrum measurement.

Conditions

Measurement view: Spectrum with the specified Measurement View Number.

The parameter <x> = 0 to 4.

Group

Calculate commands

Syntax

```
CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:POWer:INTEgrated? (QUERY)
```

Arguments

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

`<NR2>` = a floating point value without exponents. It is the integrated power reading between MR (reference marker) and the selected marker in dBm.

Examples

`CALCulate:SPECtrum:MEASview1:MARKer1:POWer:INTEgrated?` might return -12.4462022781, indicating the integrated power is -12.45 dBm.

CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>[:SET]:CENTer (No Query Form)

Sets the center frequency to the marker frequency in the spectrum measurement.

Conditions

Measurement views: Spectrum with the specified Measurement View Number

Group

Calculate commands

Syntax

```
CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>[:SET]:CENTer
```

Arguments

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

<NRf>

Examples

`CALCulate:SPECtrum:MEASview1:MARKer1:SET:CENTER` sets the center frequency to the marker frequency in the Spectrum Measurement View 1

CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:TRACe

Sets or queries the trace on which the specified marker is placed in the spectrum measurement.

Conditions

Measurement views: Spectrum with specified Measurement View

Group

Calculate commands

Syntax

```
CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:TRACe { TRACE1 | TRACE2 | TRACE3 | TRACE4 | TRACE5 }
CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:TRACe? (QUERY)
```

Arguments

`TRACE1` places the specified marker on Trace 1.

`TRACE2` places the specified marker on Trace 2.

`TRACE3` places the specified marker on n Trace 3.

`TRACE4` places the specified marker on Trace 4 (math trace).

`TRACE5` places the specified marker on Trace 5 (spectrogram).

Trace 1 to 3 can be defined as Normal, Average, Max Hold or Min Hold using the command.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

<NRf>

Examples

`CALCulate:SPECtrum:MEASview1:MARKer1:TRACe TRACe1` places Marker 1 (M1) on Trace 1 of Spectrum Measurement View.

CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:X

Sets or queries the horizontal position of the selected marker on the spectrum trace.

Conditions

Measurement views: Spectrum with the specified Measurement View Number.

Group

Calculate commands

Syntax

```
CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:X <value>
CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:X? <QUERY>
```

Arguments

<value>

Range : Start to Stop frequency (left to right edge of the horizontal axis).

Using an out-of-range value causes an execution error (-222, "Data out of range").

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

<NRf>

Examples

`CALCulate:SPECtrum:MEASview1:MARKer1:X 1GHz` places Marker 1 (M1) at 1 GHz on the trace of Spectrum Measurement View 1.

CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:Y? (Query Only)

Returns the delta marker amplitude for the selected marker on the spectrum trace

Conditions

Measurement views: Spectrum with the specified Measurement View Number

Group

Calculate commands

Syntax

```
CALCulate:SPECtrum[:MEASview<y>]:MARKer<x>:DELTa:Y? (QUERY)
```

Related Commands

CALCulate:SPECtrum:MARKer<x>:X (on page 589)

Arguments

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

`<NRf>` Delta marker amplitude for the selected marker.

Examples

`CALCulate:SPECtrum:MEASview1:MARKer1:DELTa:Y?` might return 23.45, indicating that the delta marker amplitude is 23.45 dB.

CALCulate:SPURious:MARKer<x>:DELTa:X? (Query Only)

Returns the delta marker frequency for the selected marker on the spectrum trace.

The parameter <x> = 1 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Conditions

Measurement views: Spurious

Group

Calculate commands

Syntax

```
CALCulate:SPURious:MARKer<x>:DELTa:X?
```

Arguments

None

Returns

`<NRf>` Delta marker frequency for the selected marker.

Examples

`CALCULATE:SPURIOUS:MARKER1:DELTA:X?` might return 1.28E+6, indicating that the delta marker frequency is 1.28 MHz.

CALCulate:SPURious:MARKer<x>:DELTa:Y? (Query Only)

Returns the delta marker amplitude for the selected marker on the spectrum trace.

The parameter <x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: Spurious

Group

Calculate commands

Syntax

```
CALCulate:SPURious:MARKer<x>:DELTa:Y?
```

Arguments

None

Returns

`<NRf>` Delta marker amplitude for the selected marker.

Examples

`CALCULATE:SPURIOUS:MARKER1:DELTA:Y?` might return 23.45, indicating that the delta marker amplitude is 23.45 dB.

CALCulate:SPURious:MARKer<x>:MAXimum (No Query Form)

Moves the selected marker to the highest peak on the spectrum trace.

Conditions

Measurement views: Spurious

Group

Calculate commands

Syntax

`CALCulate:SPURious:MARKer<x>:MAXimum`

Arguments

None

Examples

`CALCULATE:SPURIOUS:MARKER1:MAXIMUM` moves Marker 1 (M1) to the highest peak on the trace.

CALCulate:SPURious:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the selected marker to the next peak higher in amplitude on the spectrum trace.

Conditions

Measurement views: Spurious

Group

Calculate commands

Syntax

`CALCulate:SPURious:MARKer<x>:PEAK:HIGHer`

Arguments

None

Examples

`CALCULATE:SPURIOUS:MARKER1:PEAK:HIGHER` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

CALCulate:SPURious:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the selected marker to the next peak to the left on the spectrum trace.

Conditions

Measurement views: Spurious

Group

Calculate commands

Syntax

`CALCulate:SPURious:MARKer<x>:PEAK:LEFT`

Arguments

None

Examples

`CALCULATE:SPURIOUS:MARKER1:PEAK:LEFT` moves Marker 1 (M1) to the next peak to the left on the trace.

CALCulate:SPURious:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the selected marker to the next peak lower in amplitude on the spectrum trace.

Conditions

Measurement views: Spurious

Group

Calculate commands

Syntax

`CALCulate:SPURious:MARKer<x>:PEAK:LOWer`

Arguments

None

Examples

`CALCULATE:SPURIOUS:MARKER1:PEAK:LOWER` moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

CALCulate:SPURious:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the selected marker to the next peak to the right on the spectrum trace.

Conditions

Measurement views: Spurious

Group

Calculate commands

Syntax

`CALCulate:SPURious:MARKer<x>:PEAK:RIGHt`

Arguments

None

Examples

`CALCULATE:SPURIOUS:MARKER1:PEAK:RIGHT` moves Marker 1 (M1) to the next peak to the right on the trace.

CALCulate:SPURious:MARKer<x>[:SET]:CENTer (No Query Form)

Sets the center frequency to the marker frequency in the Spurious measurement.

Conditions

Measurement views: Spurious

Group

Calculate commands

Syntax

```
CALCulate:SPURious:MARKer<x>[:SET]:CENTer
```

Arguments

None

Examples

`CALCULATE:SPURIOUS:MARKER1:SET:CENTER` sets the center frequency to the value at Marker 1.

CALCulate:SPURious:MARKer<x>:X

Sets or queries the horizontal position of the selected marker on the spectrum trace.

Conditions

Measurement views: Spurious

Group

Calculate commands

Syntax

```
CALCulate:SPURious:MARKer<x>:X <value>
CALCulate:SPURious:MARKer<x>:X?
```

Arguments

`<value>::=<NRf>` specifies the horizontal position of the marker.

Range: Start to Stop frequency (left to right edge of the horizontal axis).

The start and stop frequencies are set using the `DISPlay:SPURious:X[:SCALe]:STARt` and `DISPlay:SPURious:X[:SCALe]:STOP` commands.

Examples

`CALCULATE:SPURIOUS:MARKER1:X 800MHz` places Marker 1 (M1) at 800 MHz on the spectrum trace.

CALCulate:SPURious:MARKer<x>:Y? (Query Only)

Queries the marker amplitude of the selected marker on the spectrum trace.

Conditions

Measurement views: Spurious

Group

Calculate commands

Syntax

```
CALCulate:SPURious:MARKer<x>:Y?
```

Arguments

None

Returns

<NRf> Marker amplitude of the selected marker.

Examples

CALCULATE:SPURIOUS:MARKER1:Y? might return -34.28, indicating Marker 1 (M1) is at -34.28 dBm.

CALCulate:SYNC:MARKer<x>:MEASurement

Sets or queries the selected measurement to which the specified marker will be snapped to the peak of the measurement trace.

Conditions

Measurement views: General signal viewing

Group

Calculate commands

Syntax

```
CALCulate:SYNC:MARKer<x>:MEASurement <value>
CALCulate:SYNC:MARKer<x>:MEASurement?
```

Arguments

<x> parameter : 0 to 4; The specified marker must be activated using the CALCulate:MARKer:ADD command.

where

<value> :: {Spectrum : Source 1 : Meas1 | Time Overview : Source 1 : Meas1} etc.

Examples

CALCulate:SYNC:MARKer0:MEASurement? queries the Selected measurement for Auto Sync MR.

CALCulate:SYNC:MARKer1:MEASurement "Time Overview : Source 1 : Meas1" sets the M1 to snap to peak of Time Overview Source 1 Measurement.

CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:DELTa:X[:TIME]? (Query only)

Returns the delta marker time for the selected marker on the Trellis diagram trace.

The parameter <x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Trellis diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Trellis diagram).

Group

Calculate commands

Syntax

CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:DELTa:X[:TIME]?

Related Commands

CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:DELTa:Y? (on page 596)

Arguments

None

Returns

<NRf> Delta marker time for the selected marker.

Use the [SENSe]:DDEMod:TIMe:UNITs command to select the time unit: symbols (default) or seconds.

Examples

CALCulate:TDIagram:MEASview2:MARKer1:DELTa:X:TIME? might return 62.75,indicating that the delta marker time is 62.75 symbols.

CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:DELTa:Y? (Query only)

Returns the delta marker amplitude for the selected marker on the Trellis diagram trace.

The parameter <x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Trellis diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Trellis diagram).

Group

Calculate commands

Syntax

CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:DELTa:Y?

Related Commands

CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:DELTa:X[:TIME]? (on page 595)

Arguments

None

Returns

<NRf> Delta marker amplitude for the selected marker in degrees.

Examples

CALCulate:TDIagram:MEASview2:MARKer1:DELTa:Y? might return -48.26, indicating that the delta marker amplitude is -48.26°.

CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:MAXimum (No query form)

Moves the selected marker to the highest peak on the Trellis diagram trace.

The parameter <x> = 0 to 4. The specified marker must be activated using `CALCulate:MARKer:ADD` command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Trellis diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Trellis diagram).

Group

Calculate commands

Syntax

```
CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:MAXimum
```

Arguments

None

Examples

`CALCulate:TDIagram:MEASview1:MARKer1:MAXimum` moves Marker 1 (M1) to the highest peak on the trace.

CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (No query form)

Moves the selected marker to the next peak higher in amplitude on the Trellis diagram trace.

The parameter <x> = Marker 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Trellis diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Trellis diagram)

Group

Calculate commands

Syntax

```
CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:PEAK:HIGHer
```

Related Commands

CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:PEAK:LOWer (on page 598)

Arguments

None

Examples

`CALCulate:TDIagram:MEASview2:MARKer1:PEAK:HIGHer` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:PEAK:LEFT (No query form)

Moves the selected marker to the next peak to the left on the Trellis diagram trace.

The parameter <x> = Marker 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Trellis diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Trellis diagram).

Group

Calculate commands

Syntax

`CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:PEAK:LEFT`

Related Commands

CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (on page 599)

Arguments

None

Examples

`CALCulate:TDIagram:MEASview2:MARKer1:PEAK:LEFT` moves Marker 1 (M1) to the next peak to the left on the trace.

CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:PEAK:LOWer (No query form)

Moves the selected marker to the next peak lower in amplitude on the Trellis diagram trace.

The parameter <x> = Marker 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Trellis diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Trellis diagram).

Group

Calculate commands

Syntax

`CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:PEAK:LOWer`

Related Commands

CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (on page 597)

Arguments

None

Examples

`CALCulate:TDIagram:MEASview2:MARKer1:PEAK:LOWer` moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (No query form)

Moves the selected marker to the next peak to the right on the Trellis diagram trace.

The parameter <x> = Marker 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Trellis diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Trellis diagram).

Group

Calculate commands

Syntax

`CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:PEAK:RIGHt`

Related Commands

CALCulate:TDIagram [:MEASview<y>]:MARKer<x>:PEAK:LEFT (on page 598)

Arguments

None

Examples

`CALCulate:TDIagram:MEASview2:MARKer1:PEAK:RIGHt` moves Marker 1 (M1) to the next peak to the right on the trace.

CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:X[:TIME]

Sets or queries the horizontal position (time) of the selected marker in the Trellis diagram measurement.

The parameter <x> = Marker 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Trellis diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Trellis diagram).

Group

Calculate commands

Syntax

```
CALCulate:TDIagram [:MEASview<y>]:MARKer<x>:X[:TIME] <value>
CALCulate:TDIagram:MARKer<x>:X[:TIME]?
```

Related Commands

CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:Y? (on page 600)

Arguments

<value>::=<NRf> specifies the horizontal position (time) of the marker. Use the [SENSe]:DDEMod:TIMe:UNITs command to select the time unit as symbols (default) or seconds.

Examples

CALCulate:TDIagram:MEASview2:MARKer1:X:TIME 38.5 places Marker 1 (M1) at 38.5 symbols on the trace.

CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:Y? (Query only)

Queries the vertical position of the selected marker in the Trellis diagram measurement.

The parameter <x> = Marker 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Trellis diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Trellis diagram).

Group

Calculate commands

Syntax

```
CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:Y?
```

Related Commands

CALCulate:TDIagram[:MEASview<y>]:MARKer<x>:X[:TIME] (on page 599)

Arguments

None

Returns

<NRf> Marker amplitude of the selected marker in degrees.

Examples

CALCulate:TDIagram:MEASview2:MARKer1:Y? might return 212.3, indicating Marker 1 (M1) is at 212.3 °.

CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:DELTa:X? (Query Only)

Returns the delta marker time for the selected marker on the time overview trace.

Conditions

Measurement views: Time overview with specified Measurement view number

Group

Calculate commands

Syntax

CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:DELTa:X?

Related Commands

CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:DELTa:Y? (on page 601)

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

The parameter <x> = 1 to 4.

Returns

<NRf> Delta marker time for the selected marker.

Examples

CALCULATE:TOVERVIEW:MEASVIEW1:MARKER1:DELTA:X? might return 38.0E-9, indicating that the delta marker time is 38.0 ns.

CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:DELTa:Y? (Query Only)

Returns the delta marker amplitude for the selected marker on the time overview trace.

The parameter <x> = 1 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: Time overview with specified Measurement view number

Group

Calculate commands

Syntax

```
CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:DELTa:Y?
```

Related Commands

CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:DELTa:X? (on page 601)

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

The parameter <x> = 1 to 4.

Returns

<NRf> Delta marker amplitude for the selected marker.

Examples

CALCULATE:TOVERVIEW:MEASVIEW1:MARKER1:DELTA:Y? might return 23.45, indicating that the delta marker amplitude is 23.45 dB on the trace of Time overview Measurement View 1.

CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:MAXimum (No Query Form)

Moves the selected marker to the highest peak on the time overview trace.

Conditions

Measurement views: Time overview displays with specified Measurement view number

Group

Calculate commands

Syntax

```
CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:MAXimum
```

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Examples

CALCULATE:TOVERVIEW:MEASview1:MARKer1:MARKER1:MAXIMUM moves Marker 1 (M1) to the highest peak on the trace.

CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the selected marker to the next peak higher in amplitude on the time overview trace.

Conditions

Measurement views: Time overview with specified Measurement view number

Group

Calculate commands

Syntax

```
CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:PEAK:HIGHer
```

Related Commands

CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:PEAK:LOWer (on page 603)

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

The parameter <x> = 0 to 4.

Examples

`CALCULATE:TOVERVIEW:MEASVIEW1:MARKER1:PEAK:HIGHER` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the selected marker to the next peak to the left on the time overview trace.

Conditions

Measurement views: Time overview with specified Measurement view number

Group

Calculate commands

Syntax

```
CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:PEAK:LEFT
```

Related Commands

CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (on page 604)

Arguments

None

Examples

`CALCULATE:TOVERVIEW:MEASVIEW1:MARKER1:PEAK:LEFT` moves Marker 1 (M1) to the next peak to the left on the trace.

CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the selected marker to the next peak lower in amplitude on the time overview trace.

Conditions

Measurement views: Time overview with specified Measurement view number

Group

Calculate commands

Syntax

```
CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:PEAK:LOWer
```

Related Commands

CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:PEAK:HIGHer (on page 602)

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

The parameter <x> = 0 to 4.

Examples

`CALCULATE:TOVERVIEW:MEASVIEW1:MARKER1:PEAK:LOWER` moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the selected marker to the next peak to the right on the time overview trace.

Conditions

Measurement views: Time overview with specified Measurement view number

Group

Calculate commands

Syntax

```
CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:PEAK:RIGHt
```

Related Commands

CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:PEAK:LEFT (on page 603)

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

The parameter <x> = 0 to 4.

Examples

`CALCULATE:TOVERVIEW:MEASVIEW1:MARKER1:PEAK:RIGHT` moves Marker 1 (M1) to the next peak to the right on the trace.

CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:X

Sets or queries the horizontal position of the selected marker on the time overview trace.

Conditions

Measurement views: Time overview with specified Measurement view number

Group

Calculate commands

Syntax

```
CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:X <value>
CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:X?
```

Related Commands

CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:Y? (on page 605)

Arguments

`<value>::=<NRf>` specifies the horizontal position of the specified marker.

Range: (analysis offset) to [(analysis offset) + (analysis length)].

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

The parameter <x> = 0 to 4.

Returns

`<NRf>` is a floating point number that is the horizontal position of the specified marker.

Examples

`CALCULATE:TOVERVIEW:MEASVIEW1:MARKER1:X 1.5us` places Marker 1 (M1) at 1.5 µs on the trace.

CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:Y? (Query Only)

Queries the marker amplitude of the selected marker in the time overview.

Conditions

Measurement views: Time overview with specified Measurement view number

Group

Calculate commands

Syntax

```
CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:Y?
```

Related Commands

CALCulate:TOVerview[:MEASview<y>]:MARKer<x>:X (on page 604)

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

The parameter <x> = 0 to 4.

Returns

`<NRf>` Marker amplitude of the selected marker.

Examples

`CALCULATE:TOVERVIEW:MEASVIEW1:MARKER1:Y?` might return `-34.28`, indicating Marker 1 (M1) is at −34.28 dBm.

CALculate:TXGain:MARKer<x>:DELTa:X? (Query Only)

Returns the delta marker frequency for the specified marker on the trace.

Conditions

Measurement view: Transmission Gain

The parameter <x> = 1, 2, 3, 4 is the marker. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Group

Calculate commands

Syntax

`CALculate:TXGain:MARKer<x>:DELTa:X?`

Returns

`<NRf>` is the delta marker frequency for the specified marker.

Examples

`CALculate:TXGain:MARKer1:DELTa:X?` might return 1.28E+6, indicating that the delta marker frequency is 1.28 MHz.

CALculate:TXGain:MARKer<x>:DELTa:Y? (Query Only)

Returns the delta marker amplitude for the specified marker on the trace.

Conditions

Measurement view: Transmission Gain

The parameter <x> = 1 to 4 is the marker. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Group

Calculate commands

Syntax

`CALculate:TXGain:MARKer<x>:DELTa:Y?`

Returns

`<NRf>` is the delta marker amplitude for the specified marker.

Examples

`CALculate:TXGain:MARKer1:DELTa:Y?` might return `23.45`, indicating that the delta marker amplitude for Marker 1 (M1) is 23.45 dB.

CALculate:TXGain:MARKer<x>:MAXimum (No Query Form)

Moves the specified marker to the highest peak on the trace.

Conditions

Measurement view: Transmission Gain

The parameter <x> = 0,1,2,3,4 is the marker. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Group

Calculate commands

Syntax

`CALculate:TXGain:MARKer<x>:MAXimum`

Arguments

None

Examples

`CALculate:TXGain:MARKer1:MAXimum` moves Marker 1 (M1) to the highest peak on the trace.

CALculate:TXGain:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the specified marker to the next peak higher in amplitude on the trace.

Conditions

Measurement view: Transmission Gain

The parameter <x> = 0,1,2,3,4 is the marker. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Group

Calculate commands

Syntax

`CALculate:TXGain:MARKer<x>:PEAK:HIGHer`

Arguments

None

Examples

`CALculate:TXGain:MARKer1:HIGHer` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

CALculate:TXGain:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the specified marker to the next peak to the left on the trace.

Conditions

Measurement view: Transmission Gain

The parameter <x> = 0,1,2,3,4 is the marker. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Group

Calculate commands

Syntax

```
CALculate:TXGain:MARKer<x>:PEAK:LEFT
```

Arguments

None

Examples

`CALculate:TXGain:MARKer1:PEAK:LEFT` moves Marker 1 (M1) to the next peak to the left on the trace.

CALculate:TXGain:MARKer<x>:PEAK:LOWer (No Query Form)

-Moves the specified marker to the next peak lower in amplitude on the trace.

Conditions

Measurement view: Transmission Gain

The parameter <x> = 0,1,2,3,4 is the marker. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Group

Calculate commands

Syntax

```
CALculate:TXGain:MARKer<x>:PEAK:LOWer
```

Arguments

None

Examples

`CALculate:TXGain:MARKer1:PEAK:LOWer` moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

CALculate:TXGain:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the specified marker to the next peak to the right on the trace.

Conditions

Measurement view: Transmission Gain

The parameter <x> = 0,1,2,3,4 is the marker. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Group

Calculate commands

Syntax

```
CALculate:TXGain:MARKer<x>:PEAK:RIGHt
```

Arguments

None

Examples

`CALculate:TXGain:MARKer1:PEAK:RIGHt` moves Marker 1 (M1) to the next peak to the right on the trace.

CALculate:TXGain:MARKer<x>:TRACe

Sets or queries the trace on which the specified marker is placed in the Transmission Gain display.

Conditions

Measurement view: Transmission Gain

The parameter <x> = 0,1,2,3,4 is the marker. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Group

Calculate commands

Syntax

```
CALculate:TXGain:MARKer<x>:TRACe { TRACE1 | TRACE2 | TRACE3 }
CALculate:TXGain:MARKer<x>:TRACe?
```

Arguments

`TRACE1` places the specified marker on Trace 1.

`TRACE2` places the specified marker on Trace 2.

`TRACE3` places the specified marker on Trace 3.

Returns

See Arguments.

Examples

`CALculate:TXGain:MARKer1:TRACe TRACE1` places Marker 1 (M1) on Trace 1.

CALculate:TXGain:MARKer<x>:X

Sets or queries the horizontal position for the specified marker on the trace.

Conditions

Measurement view: Transmission Gain

The parameter <x> = 0,1,2,3,4 is the marker. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Group

Calculate commands

Syntax

```
CALculate:TXGain:MARKer<x>:X <value>
CALculate:TXGain:MARKer<x>:X ?
```

Arguments

`<value>::=<NRf>` specifies the horizontal position of the specified marker.

Range: Start to Stop frequency (left to right edge of the horizontal axis). Using an out-of-range value causes an execution error (-222, "Data out of range").

Returns

<NRf> is the horizontal position of the specified marker.

Examples

CALculate:TXGain:MARKer1:X 800E+6 places Marker 1 (M1) at 800 MHz on the trace.

CALculate:TXGain:MARKer<x>:Y? (Query Only)

Queries the marker amplitude of the specified marker on the trace.

Conditions

Measurement view: Transmission Gain

The parameter <x> = 0,1,2,3,4 is the marker. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Group

Calculate commands

Syntax

CALculate:TXGain:MARKer<x>:Y?

Returns

<NRf> is the amplitude at the specified marker position in dB.

Examples

CALculate:TXGain:MARKer1:Y might return -23.45, indicating that the marker amplitude is -23.45 dB.

CALCulate:WLAN:AD:CONSte:MARKer<x>:DELTa:X[:TIME]? (Query Only)

Returns the time of the specified delta marker on the trace in the Constellation display.

Conditions

Measurement view: 802.11ad Constellation

Group

Calculate commands

Syntax

CALCulate:WLAN:AD:CONSte:MARKer<x>:DELTa:X[:TIME]?

Arguments

<x> = 1,2,3,4 is the marker.

Returns

<NR3>

Examples

CALCulate:WLAN:AD:CONSte:MARKer1:DELTa:X:TIME? might return 3.1494-6, indicating that the frequency of the specified delta between Marker (M1) and MR is 3.1494-6.

CALCulate:WLAN:AD:CONSte:MARKer<x>:MAGNitude? (Query Only)

Queries the marker magnitude of the specified marker in the Constellation display.

Conditions

Measurement view: 802.11ad Constellation

Group

Calculate commands

Syntax

`CALCulate:WLAN:AD:CONSte:MARKer<x>:MAGNitude?`

Arguments

`<x>` = 0,1,2,3,4 is the marker.

Returns

`<NRf>` The magnitude for the selected marker

Examples

`CALCulate:WLAN:AD:CONSte:MARKer1:MAGNitude?` might return 6.261E-3, indicating the magnitude of Marker 1(M1) is 6.261E-3.

CALCulate:WLAN:AD:CONSte:MARKer<x>:PHASe? (Query Only)

Queries the phase readout of the specified marker in the Constellation display.

Conditions

Measurement view: 802.11ad Constellation

Group

Calculate commands

Syntax

`CALCulate:WLAN:AD:CONSte:MARKer<x>:PHASe?`

Arguments

`<x>` = 0,1,2,3,4 is the marker.

Returns

`<NRf>` The phase readout for the selected marker

Examples

`CALCulate:WLAN:AD:CONSte:MARKer1:PHASe?` might return 9.690, indicating the phase readout of Marker 1 (M1) is 9.7°.

CALCulate:WLAN:AD:CONSte:MARKer<x>:SYMBol? (Query Only)

Queries the symbol of the specified marker in Constellation display.

Conditions

Measurement view: 802.11ad Constellation

Group

Calculate commands

Syntax

`CALCulate:WLAN:AD:CONSte:MARKer<x>:SYMBol?`

Arguments

`<x>` = 0,1,2,3,4 is the marker.

Returns

`<NR3>` = the symbol of the selected marker

Examples

`CALCulate:WLAN:AD:CONSte:MARKer1:SYMBol?` might return 3.369E+3, indicating the symbol value of the Marker 1 (M1) is 3.369E+3.

CALCulate:WLAN:AD:CONSte:MARKer<x>:TYPE? (Query Only)

Queries the data type of the specified marker in Constellation display.

Conditions

Measurement view: 802.11ad Constellation

Group

Calculate commands

Syntax

`CALCulate:WLAN:AD:CONSte:MARKer<x>:TYPE?`

Arguments

`<x>` = 0,1,2,3,4 is the marker.

Returns

`<String>`

Either (Preamble | Header | Data | Guard)

Examples

`CALCulate:WLAN:AD:CONSte:MARKer1:TYPE?` might return `"Preamble"`, indicating the data type of the selected marker is “Preamble".

CALCulate:WLAN:AD:CONSte:MARKer<x>:VALue? (Query Only)

Queries the value of the selected marker in Constellation display.

Conditions

Measurement view: 802.11ad Constellation

Group

Calculate commands

Syntax

```
CALCulate:WLAN:AD:CONSte:MARKer<x>:VALue?
```

Arguments

<x> = 0,1,2,3,4 is the marker.

Returns

<NR1> = the value of the selected marker

Examples

CALCulate:WLAN:AD:CONSte:MARKer1:VALue? might return "+1", indicating the value of the Marker 1 (M1) is "+1".

CALCulate:WLAN:AD:CONSte:MARKer<x>:X

Sets or queries the horizontal position of the specified marker on the trace in the 802.11ad Constellation display.

Conditions

Measurement view: 802.11ad Constellation

<x> = 0,1,2,3,4 is the marker.

Group

Calculate commands

Syntax

```
CALCulate:WLAN:AD:CONSte:MARKer<x>:X <value>
CALCulate:WLAN:AD:CONSte:MARKer<x>:X?
```

Arguments

<value>=<NRf> specifies the horizontal position of the marker

Returns

<NRf>

Examples

CALCulate:WLAN:AD:CONSte:MARKer1:X 9.5409090173E-6 places Marker 1 (M1) at 9.540E−6 on the trace.

CALCulate:WLAN:AD:EVM:MARKer<x>:DELTa:X? (Query Only)

Returns the delta marker time for the selected marker on the EVM trace.

Conditions

Measurement views: 802.11ad EVM vs Time

Group

Calculate commands

Syntax
CALCulate:WLAN:AD:EVM:MARKer<x>:DELTa:X?

Arguments
<x> = 1,2,3,4 is the marker.

Returns
<NRf> Delta marker time

Examples
CALCulate:WLAN:AD:EVM:MARKer1:DELTa:X? might return 1.28E-9, indicating that the delta marker time for Marker 1 (M1) is 1.28E-9 s.

CALCulate:WLAN:AD:EVM:MARKer<x>:DELTa:Y? (Query Only)
Returns the delta marker amplitude for the selected marker on the EVM trace.

Conditions
Measurement views: 802.11ad EVM vs Time

Group
Calculate commands

Syntax
CALCulate:WLAN:AD:EVM:MARKer<x>:DELTa:Y?

Arguments
<x> = 1,2,3,4 is the marker.

Returns
<NRf> = Delta marker amplitude

Examples
CALCulate:WLAN:AD:EVM:MARKer<x>:DELTa:Y? might return 23.45, indicating that the delta marker amplitude for Marker 1 (M1) is 23.45 dB.

CALCulate:WLAN:AD:EVM:MARKer<x>:MAXimum (No Query Form)
Moves the selected marker to the highest peak on the trace in the EVM measurement.

Conditions
Measurement views: 802.11ad EVM vs Time

Group
Calculate commands

Syntax
CALCulate:WLAN:AD:EVM:MARKer<x>:MAXimum

Arguments

<x> = 0,1,2,3,4 is the marker.

Returns

None

Examples

`CALCulate:WLAN:AD:EVM:MARKer1:MAXimum` moves Marker 1 (M1) to the highest peak on the trace.

CALCulate:WLAN:AD:EVM:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the selected marker to the next peak higher in amplitude on the EVM trace.

Conditions

Measurement views: 802.11ad EVM vs Time

Group

Calculate commands

Syntax

`CALCulate:WLAN:AD:EVM:MARKer<x>:PEAK:HIGHer`

Arguments

`<x>` = 0,1,2,3,4 is the marker.

Returns

None

Examples

`CALCulate:WLAN:AD:EVM:MARKer1:PEAK:HIGHer` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

CALCulate:WLAN:AD:EVM:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the selected marker to the next significant peak found to the left of the current marker position.

Conditions

Measurement views: 802.11ad EVM vs Time

Group

Calculate commands

Syntax

`CALCulate:WLAN:AD:EVM:MARKer<x>:PEAK:LEFT`

Arguments

`<x>`= 0, 1,2,3,4 is the marker

Returns

None

Examples

`CALCulate:WLAN:AD:EVM:MARKer1:PEAK:LEFT` moves Marker 1 (M1) to the next peak found on the left side of the trace.

CALCulate:WLAN:AD:EVM:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the selected marker to the next peak lower in amplitude on the EVM trace.

Conditions

Measurement views: 802.11ad EVM vs Time

Group

Calculate commands

Syntax

`CALCulate:WLAN:AD:EVM:MARKer<x>:PEAK:LOWer`

Arguments

`<x>` = 0,1,2,3,4 is the marker.

Returns

None

Examples

`CALCulate:WLAN:AD:EVM:MARKer1:PEAK:LOWer` moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

CALCulate:WLAN:AD:EVM:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the selected marker to the next significant peak found to the right of the current marker position.

Conditions

Measurement views: 802.11ad EVM vs Time

Group

Calculate commands

Syntax

`CALCulate:WLAN:AD:EVM:MARKer<x>:PEAK:RIGHt`

Arguments

`<x>` = 0,1,2,3,4 is the marker.

Returns

None

Examples

`CALCulate:WLAN:AD:EVM:MARKer1:PEAK:RIGHt` moves Marker 1 (M1) to the next peak found on the right side of the trace.

CALCulate:WLAN:AD:EVM:MARKer<x>:X

Sets or queries the horizontal (time) stamp of the selected marker on the EVM trace.

Conditions

Measurement views: 802.11ad EVM vs Time

Group

Calculate commands

Syntax

```
CALCulate:WLAN:AD:EVM:MARKer<x>:X <value>
CALCulate:WLAN:AD:EVM:MARKer<x>:X?
```

Arguments

where

`<x>` = 0,1,2,3,4 is the marker.

`<value>= <NRf>`

Returns

`<NRf>`

Examples

`CALCulate:WLAN:AD:EVM:MARKer<x>:PEAK:LOWer 5.36` places Marker 1 (M1) at 5.36 µs on the EVM trace.

CALCulate:WLAN:AD:EVM:MARKer<x>:Y? (Query Only)

Queries the vertical position of the selected marker on the EVM trace.

Conditions

Measurement views: 802.11ad EVM vs Time

Group

Calculate commands

Syntax

```
CALCulate:WLAN:AD:EVM:MARKer<x>:Y?
```

Arguments

`<x>` = 0,1,2,3,4 is the marker.

Returns

`<NRf>` Vertical position of marker

Examples

CALCulate:WLAN:AD:EVM:MARKer1:Y? might return -34.28, indicating the vertical position of Marker 1 (M1) is -34.28 dBm on the EVM trace.

CALCulate:WLAN:AY:CONSte:MARKer<x>:DELTa:X:[:TIME]? (Query Only)

Query the value of the specified delta marker on the trace in the Constellation display.

You must activate the specified marker by using CALCulate:MARKer:ADD.

Conditions

Measurement view: 802.11ay Analysis Constellation

<x> = 1 to 4.

Group

Calculate commands

Syntax

CALCulate:WLAN:AY:CONSte:MARKer<x>:DELTa:X:[:TIME]?

Returns

<NRf>

Units: s (seconds)

Examples

CALC:WLAN:AY:CONS:MARK1:DELT:X:TIME? may return -1.772e-6, indicating that -1.772e-6 seconds is value of the specified delta marker between Marker 1 and the Reference Marker.

CALCulate:WLAN:AY:CONSte:MARKer<x>:MAGNitude? (Query Only)

Query the marker magnitude of the specified marker in the Constellation display.

Conditions

Measurement view: 802.11ay Analysis Constellation

<x> = 0 to 4.

Group

Calculate commands

Syntax

CALCulate:WLAN:AY:CONSte:MARKer<x>:MAGNitude?

Returns

<NRf> is the magnitude for the specified delta marker.

Examples

CALC:WLAN:AY:CONS:MARK2:MAGN? may return 6.261e-3, indicating that the magnitude for Marker 2 is 6.261e-3.

CALCulate:WLAN:AY:CONSte:MARKer<x>:PHASe? (Query Only)

Query the phase readout of the specified marker in the Constellation display.

Conditions

Measurement view: 802.11ay Analysis Constellation

<x> = 0 to 4.

Group

Calculate commands

Syntax

```
CALCulate:WLAN:AY:CONSte:MARKer<x>:PHASe?
```

Returns

<NRf> is the phase readout for the specified marker.Units: degrees

Examples

CALC:WLAN:AY:CONS:MARK3:PHAS? may return 9.690, indicating that the phase readout for Marker 3 is 9.7° degrees.

CALCulate:WLAN:AY:CONSte:MARKer<x>:SYMBol? (Query Only)

Query the Symbol value at which the specified marker is placed.

Conditions

Measurement view: 802.11ay Analysis Constellation

<x> = 0 to 4.

Group

Calculate commands.

Syntax

```
CALCulate:WLAN:AY:CONSte:MARKer<x>:SYMBol?
```

Returns

<NR1>

Examples

CALC:WLAN:AY:CONS:MARK1:SYMB? may return 3.369e+3, indicating that the symbol value at which Marker 1 is placed is 3.369e+3.

CALCulate:WLAN:AY:CONSte:MARKer<x>:TYPE? (Query Only)

Query the data type of the specified marker in the Constellation display.

The data type can be Preamble, Header, Data, or Guard, based on the marker position selection in the region.

Conditions

Measurement view: 802.11ay Analysis Constellation

<x> = 0 to 4.

Group

Calculate commands

Syntax

```
CALCulate:WLAN:AY:CONSte:MARKer<x>:TYPE?
```

Returns

<string>

Examples

CALC:WLAN:AY:CONS:MARK1:TYPE? may return Preamble, indicating that the data type for Marker 1 is Preamble.

CALCulate:WLAN:AY:CONSte:MARKer<x>:VALue? (Query Only)

Query the symbol value at which the marker is placed.

Conditions

Measurement view: 802.11ay Analysis Constellation

<x> = 0 to 4.

Group

Calculate commands

Syntax

```
CALCulate:WLAN:AY:CONSte:MARKer<x>:VALue?
```

Returns

<string>

Examples

CALC:WLAN:AY:CONS:MARK1:VAL? may return +1, which indicates the symbol value at which Marker 1 is placed.

CALCulate:WLAN:AY:CONSte:MARKer<x>:X

Set or query the horizontal position of the specified marker on the trace in the Constellation display.

Conditions

Measurement view: 802.11ay Analysis Constellation

<x> = 0 to 4.

Group

Calculate commands

Syntax

```
CALCulate:WLAN:AY:CONSte:MARKer<x>:X <value>
CALCulate:WLAN:AY:CONSte:MARKer<x>:X?
```

Arguments

<value>::=<NRf> specifies the horizontal position of the specified marker.

Range: 0.1 to 10 s (seconds)

Returns

<NRf>

Examples

CALC:WLAN:AY:CONS:MARK1:X 9.5409090173e-6 places the horizontal position of Marker 1 at 9.540e-6 on the trace.

CALC:WLAN:AY:CONS:MARK2:X? may return 9.540e-6, which is the horizontal position of Marker 2 on the trace.

CALCulate:WLAN:AY:EVM:MARKer<x>:DELTa:X

Set or query the delta marker value for the specified marker on the EVM trace.

Conditions

Measurement views: 802.11ay Analysis EVM

<x> = 1 to 4.

You must activate the specified marker by using CALCulate:MARKer:ADD.

Group

Calculate commands

Syntax

```
CALCulate:WLAN:AY:EVM:MARKer<x>:DELTa:X <NRf>
CALCulate:WLAN:AY:EVM:MARKer<x>:DELTa:X?
```

Related Commands

CALCulate:MARKer:ADD (on page 458)

Arguments

<NRf> is the delta marker value for the specified marker.

Units: s (seconds)

Returns

<NRf>

Examples

CALC:WLAN:AY:EVM:MARK1:DELT:X 1.28e-9 sets the delta marker value for Marker 1 to 1.28e-9 s.

CALC:WLAN:AY:EVM:MARK2:DELT:X? may return 1.28e-9, indicating that the delta marker value for Marker 2 is 1.28e-9 s.

CALCulate:WLAN:AY:EVM:MARKer<x>:DELTa:Y

Set or query the delta marker amplitude for the specified marker on the EVM trace.

Conditions

Measurement views: 802.11ay Analysis EVM

<x> = 1 to 4.

You must activate the specified marker by using CALCulate:MARKer:ADD.

Group

Calculate commands

Syntax

```
CALCulate:WLAN:AY:EVM:MARKer<x>:DELTa:Y <NRf>
CALCulate:WLAN:AY:EVM:MARKer<x>:DELTa:Y?
```

Related Commands

CALCulate:MARKer:ADD (on page 458)

Arguments

<NRf> is the delta marker amplitude for the specified marker.

Units: dB

Returns

<NRf>

Examples

CALC:WLAN:AY:EVM:MARK1:DELT:Y 23.45 sets the delta marker amplitude of Marker 1 to 23.45 dB.

CALC:WLAN:AY:EVM:MARK2:DELT:Y? may return 23.45, which means the marker amplitude of Marker 2 is 23.45 dB.

CALCulate:WLAN:AY:EVM:MARKer<x>:MAXimum (No Query Form)

Move the specified marker to the highest peak on the trace in the EVM display.

Conditions

Measurement views: 802.11ay Analysis EVM

<x> = 0 to 4.

You must activate the specified marker by using CALCulate:MARKer:ADD.

Group

Calculate commands

Syntax

```
CALCulate:WLAN:AY:EVM:MARKer<x>:MAXimum
```

Related Commands

CALCulate:MARKer:ADD (on page 458)

Arguments

None

Examples

CALC:WLAN:AY:EVM:MARK1:MAX moves Marker 1 to the highest peak on the trace.

CALCulate:WLAN:AY:EVM:MARKer<x>:PEAK:HIGHer (No Query Form)

Move the specified marker to the next peak higher in amplitude on the EVM trace.

Conditions

Measurement views: 802.11ay Analysis EVM

<x> = 0 to 4.

You must activate the specified marker by using CALCulate:MARKer:ADD.

Group

Calculate commands

Syntax

```
CALCulate:WLAN:AY:EVM:MARKer<x>:PEAK:HIGHer
```

Related Commands

CALCulate:MARKer:ADD (on page 458)

Arguments

None

Examples

`CALC:WLAN:AY:EVM:MARK1:PEAK:HIGH` moves Marker 1 to the next peak higher in amplitude on the trace.

CALCulate:WLAN:AY:EVM:MARKer<x>:PEAK:LEFT (No Query Form)

-Move the specified marker to the next significant peak found to the left of the current marker position.

Conditions

Measurement views: 802.11ay Analysis EVM

<x> = 0 to 4.

You must activate the specified marker by using CALCulate:MARKer:ADD.

Group

Calculate commands

Syntax

```
CALCulate:WLAN:AY:EVM:MARKer<x>:PEAK:LEFT
```

Related Commands

CALCulate:MARKer:ADD (on page 458)

Arguments

None

Examples

`CALC:WLAN:AY:EVM:MARK4:PEAK:LEFT` moves Marker 4 to the next peak to the left on the trace.

CALCulate:WLAN:AY:EVM:MARKer<x>:PEAK:LOWer (No Query Form)

Move the specified marker to the next peak lower in amplitude on the EVM trace.

Conditions

Measurement views: 802.11ay Analysis EVM

<x> = 0 to 4.

You must activate the specified marker by using CALCulate:MARKer:ADD.

Group

Calculate commands

Syntax

```
CALCulate:WLAN:AY:EVM:MARKer<x>:PEAK:LOWer
```

Related Commands

CALCulate:MARKer:ADD (on page 458)

Arguments

None

Examples

`CALC:WLAN:AY:EVM:MARK2:PEAK:LOW` moves Marker 2 to the next peak lower in amplitude on the trace.

CALCulate:WLAN:AY:EVM:MARKer<x>:PEAK:RIGHt (No Query Form)

Move the specified marker to the next significant peak found to the right of the current marker position.

Conditions

Measurement views: 802.11ay Analysis EVM

<x> = 0 to 4.

You must activate the specified marker by using CALCulate:MARKer:ADD.

Group

Calculate commands

Syntax

```
CALCulate:WLAN:AY:EVM:MARKer<x>:PEAK:RIGHt
```

Related Commands

CALCulate:MARKer:ADD (on page 458)

Arguments

None

Examples

CALC:WLAN:AY:EVM:MARK1:PEAK:RIGH moves Marker 1 to the next peak to the right on the trace.

CALCulate:WLAN:AY:EVM:MARKer<x>:X

Set or query the horizontal (time) stamp of the specified marker on the EVM trace.

Conditions

Measurement views: 802.11ay Analysis EVM

<x> = 0 to 4.

You must activate the specified marker by using CALCulate:MARKer:ADD.

Group

Calculate commands

Syntax

```
CALCulate:WLAN:AY:EVM:MARKer<x>:X <value>
CALCulate:WLAN:AY:EVM:MARKer<x>:X?
```

Related Commands

CALCulate:MARKer:ADD (on page 458)

Arguments

<value>::=<NRf> is the specified marker.

Units: µs

Returns

<NRf>

Examples

CALC:WLAN:AY:EVM:MARK1:X 5e-6 places Marker 1 at 5 µs on the EVM trace.

CALC:WLAN:AY:EVM:MARK2:X? may return 5e-6, which means Marker 2 is placed at 5 µs on the EVM trace.

CALCulate:WLAN:AY:EVM:MARKer<x>:Y

Set or query the vertical (EVM) position of the specified marker on the EVM trace.

Conditions

Measurement views: 802.11ay Analysis EVM

<x> = 0 to 4.

You must activate the specified marker by using CALCulate:MARKer:ADD.

Group

Calculate commands

Syntax

```
CALCulate:WLAN:AY:EVM:MARKer<x>:Y <NRf>
CALCulate:WLAN:AY:EVM:MARKer<x>:Y?
```

Related Commands

CALCulate:MARKer:ADD (on page 458)

Arguments

<NRf> is the marker amplitude of the specified marker.

Units: dB

Returns

<NRf>

Examples

CALC:WLAN:AY:EVM:MARK1:Y -34.28 sets the position of Marker 1 at –34.28 dB on the EVM trace.

CALC:WLAN:AY:EVM:MARK2:Y? may return -34.28, indicating that Marker 2 is at –34.28 dB on the EVM trace.

CALCulate:WLAN:CONSte:MARKer<x>:FREQuency

Sets or queries the frequency value of the marker on the WLAN Constellation display. The units are either Frequency in Hz or Subcarrier number. To set the units, use the command [SENSe]:WLAN:UNIT:FREQuency.

Note that FREQuency is not an actual measurement of frequency, but is the subcarrier offset within the analyzed packet burst.

<x>, which is optional, is the desired marker to use. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Commanding a marker which does not exist results in an execution error.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

```
CALCulate:WLAN:CONSte:MARKer<x>:FREQuency <NRf>
CALCulate:WLAN:CONSte:MARKer<x>:FREQuency?
```

Arguments

Floating point number that indicates the desired subcarrier to which the marker should be moved. Subcarriers are traditionally numbered with both positive and negative offsets from the CF for the modulation scheme. Valid integral values range from –maxsubcarrier to maxsubcarrier, where the actual values depend on the modulation standard. Specifying a value of <NRf> outside of the range of subcarriers present in the burst selects the nearest valid subcarrier. Subcarriers can also be specified by frequency (also determined by modulation standard).

Returns

Returns a unitless number representing the subcarrier number.

Examples

CALC:WLAN:CONST:MARK2:FREQ 2.0000 sets the position of Marker 2 (M2) to Subcarrier 2 (assuming the units have been set to Subcarrier).

`CALC:WLAN:CONST:MARK2:FREQ?` might return 2.41575E+9, indicating that Marker 2 (M2) is positioned at 2.41575 MHz.

CALCulate:WLAN:CONSte:MARKer<x>:MAGNitude? (Query Only)

Returns the magnitude value at the indicated marker position on the WLAN Constellation display.

<x> is optional. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

`CALCulate:WLAN:CONSte:MARKer<x>:MAGNitude?`

Returns

Returns a unitless floating point number indicating the magnitude at the selected marker position on the WLAN Constellation display.

Examples

`CALC:WLAN:CONS:MARK1:MAGN?` might return 35 indicating the magnitude at Marker 1 (M1) is 35.

CALCulate:WLAN:CONSte:MARKer<x>:PHASe? (Query Only)

Returns the phase value at the indicated marker position on the WLAN Constellation display, in degrees.

<x>, which is optional, is the desired marker to query. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

`CALCulate:WLAN:CONSte:MARKer<x>:PHASe?`

Returns

Phase value of the signal in degrees between -180 and +180 as an **ASCII** encoded floating point number in standard engineering notation.

Examples

`CALC:WLAN:CONS:MARK1:PHAS?` might return 35.74, indicating the phase readout of Marker 1 (M1) is 35.74°.

CALCulate:WLAN:CONSte:MARKer<x>:TIME

Sets the time offset value at the selected marker position on the points trace on the WLAN Constellation display. The units are either Symbols or Seconds. To set/query the units, use the command [SENSe]:WLAN:UNIT:TIME.

Note that TIME is a measurement of neither absolute nor elapsed time, but symbol offset within the analyzed packet burst.

<x>, which is optional, is the desired marker. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Commanding a marker which does not exist results in an execution error.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

```
CALCulate:WLAN:CONSte:MARKer<x>:TIME <NRf>
CALCulate:WLAN:CONSte:MARKer<x>:TIME?
```

Arguments

Floating point number that indicates the 0 based symbol or time offset from the start of packet burst. Valid integral values range from 0 to the number of symbols in the burst. Specifying a value outside of the range of symbols present in the burst selects the nearest valid symbol.

Returns

Returns a unitless integer value in floating point format that represents the symbol associated with the position of the indicated marker, or the time offset.

Examples

`CALC:WLAN:CONST:MARK3:TIME -234.5` moves Marker 3 (M3) to the 362.42 µs position on the trace (assuming the units have been set to Seconds).

`CALC:WLAN:CONST:MARK3:TIME?` might return 5.0000 indicating that Marker 3 is located at symbol 5 (assuming units have been set to Symbols).

CALCulate:WLAN:CONSte:MARKer<x>:TYPE? (Query Only)

Returns the data type of the subcarrier for data at the indicated marker position - either pilot or data.

<x>, which is optional, is the desired marker. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

CALCulate:WLAN:CONSte:MARKer<x>:TYPE?

Returns

Returns the decoded value of the subcarrier type to which the marker is attached as an **ASCII** encoded floating point number in standard engineering notation.

Values are:

1.000 if the marker type is a pilot marker.

2.000 if the marker type is a data marker.

Examples

CALC:WLAN:CONS:MARK2:TYPE? might return 1.000 indicating Marker 2 (M2) is a pilot marker.

CALCulate:WLAN:CONSte:MARKer<x>:VALue? (Query Only)

Queries the value readout at the selected marker position on the points trace of the WLAN Constellation display.

<x>, which is optional, is the desired marker to query. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

CALCulate:WLAN:CONSte:MARKer<x>:VALue?

Returns

Floating point number that represents the value readout at the selected marker position.

Examples

CALC:WLAN:CONS:MARK1:VAL? might return 2.00, indicating the value readout at Marker 1 (M1) is 2.

CALCulate:WLAN:CRESponse:MARKer<x>:DELTa:X? (Query Only)

Queries the delta Subcarrier or frequency value (depending on which units have been set) at the selected marker position on the WLAN Channel Response display.

Use the command [SENSe]:WLAN:UNIT:FREQuency to set the units.

<x> is the desired marker to query. Marker 0 is not valid for the DELTa queries. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. Valid integral values range from 1 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

`CALCulate:WLAN:CRESponse:MARKer<x>:DELTa:X?`

Returns

Floating point number that represents the delta frequency value at the selected marker position.

Examples

`CALC:WLAN:CRES:MARK2:DELT:X?` might return 1.28E+6 indicating that the delta frequency value for Marker 2 (M2) is 1.28 MHz.

CALCulate:WLAN:CRESponse:MARKer<x>:DELTa:Y? (Query Only)

Queries the vertical delta value for the points trace at the selected marker position on the WLAN Channel Response display, in dB or degree.

`<x>` is the desired marker to query. Marker 0 is not valid for the DELTa queries. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. Valid integral values range from 1 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Marker 0 and the selected marker must be on the same plot (magnitude or phase); otherwise, it will result in an execution error.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

`CALCulate:WLAN:CRESponse:MARKer<x>:DELTa:Y?`

Returns

Returns the vertical delta value for the points trace in dB or degree at the selected marker position.

Examples

`CALC:WLAN:CRES:MARK3:DELT:Y?` might return `-30`, indicating a vertical delta value of −30 dB at the Marker 3 (M3) position.

CALCulate:WLAN:CRESponse:MARKer<x>:MAXimum (No Query Form)

Moves the marker to the highest peak on the trace on the WLAN Channel Response display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

`CALCulate:WLAN:CRESponse:MARKer<x>:MAXimum`

Arguments

None

Examples

`CALC:WLAN:CRES:MARK2:MAX` moves the marker to the highest peak on the trace on the WLAN Channel Response display.

CALCulate:WLAN:CRESponse:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the marker to the next peak higher in amplitude on the WLAN Channel Response display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

`CALCulate:WLAN:CRESponse:MARKer<x>:PEAK:HIGHer`

Arguments

None

Examples

`CALC:WLAN:CRES:MARK2:PEAK:HIGHer` moves Marker 2 (M2) to the next peak higher in amplitude on the trace.

CALCulate:WLAN:CRESponse:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the marker to the next peak on the left on the trace in the WLAN Channel Response display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

CALCulate:WLAN:CRESponse:MARKer<x>:PEAK:LEFT

Arguments

None

Examples

CALC:WLAN:CRES:MARK2:PEAK:LEFT moves Marker 2 (M2) to the next peak on the left on the trace in the WLAN Channel Response display.

CALCulate:WLAN:CRESponse:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the marker to the next peak lower in amplitude on the trace on the WLAN Channel Response display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

CALCulate:WLAN:CRESponse:MARKer<x>:PEAK:LOWer

Arguments

None

Examples

CALC:WLAN:CRES:MARK2:PEAK:LOW moves Marker 2 (M2) to the next peak lower on the trace.

CALCulate:WLAN:CRESponse:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the marker to the next peak to the right on the trace on the WLAN Channel Response display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

CALCulate:WLAN:CRESponse:MARKer<x>:PEAK:RIGHt

Arguments

None

Examples

`CALC:WLAN:CRES:MARK2:PEAK:RIGH` moves Marker 2 (M2) to the next peak to the next peak on the right on the trace.

CALCulate:WLAN:CRESponse:MARKer<x>:X

Sets or queries the value of the horizontal position (units in Frequency or Subcarrier) for the selected marker on the points trace in the WLAN Channel Response display.

To set the units, use the command [SENSe]:WLAN:UNIT:FREQuency.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

```
CALCulate:WLAN:CRESponse:MARKer<x>:X <NRf>
CALCulate:WLAN:CRESponse:MARKer<x>:X?
```

Arguments

Floating point number that specifies the value of the horizontal position of the marker. Range: Start to Stop frequency (left to right edge of the horizontal axis). Using an out-of-range value causes an execution error (-222, “Data out of range”).

Examples

`CALC:WLAN:CRES:MARK3:X 22` places Marker 3 at 22 Subcarrier on the trace (assuming the units have been set to Subcarrier).

CALCulate:WLAN:CRESponse:MARKer<x>:Y? (Query Only)

Returns the value of the amplitude at the selected marker position in the WLAN Channel Response measurement.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

```
CALCulate:WLAN:CRESponse:MARKer<x>:Y?
```

Returns

Floating point number that indicates the amplitude value at the selected marker position in dB or degrees.

Examples

`CALC:WLAN:CRES:MARK1:Y?` might return `33.393856E-3` indicating the amplitude of Marker 1 is 0.033 dB.

CALCulate:WLAN:EVM:MARKer<x>:AVERage:SUBCarrier? (Query Only)

Queries the EVM value at the selected marker position on the average trace on the WLAN EVM display subcarrier graph.

To query the marker value of the points trace, use the query.

<x>, which is optional, is the desired marker to query. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

```
CALCulate:WLAN:EVM:MARKer<x>:AVERage:SUBCarrier?
```

Returns

Returns a floating point number that represents the EVM value at the selected marker position on the average trace on the WLAN EVM display subcarrier graph.

Examples

`CALC:WLAN:EVM:MARK3:AVER:SUBC?` might return 16.4, indicating an EVM value of 16.4 percent at the Marker 3 (M3) position on the average trace on the WLAN EVM display subcarrier graph.

CALCulate:WLAN:EVM:MARKer<x>:AVERage:SYMBol? (Query Only)

Queries the EVM value at the selected marker position on the average trace on the WLAN EVM display symbol graph.

To query the marker value of the points trace, use the query.

<x>, which is optional, is the desired marker to query. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

```
CALCulate:WLAN:EVM:MARKer<x>:AVERage:SYMBol?
```

Returns

Returns a floating point number that represents the EVM value at the selected marker position on the average trace on the WLAN EVM display symbol graph.

Examples

`CALC:WLAN:EVM:MARK3:AVER:SYMB?` might return `14.5`, indicating an EVM value of 14.5 percent at the Marker 3 (M3) position on the average trace on the WLAN EVM display symbol graph.

CALCulate:WLAN:EVM:MARKer<x>:DELTa:SUBCarrier? (Query Only)

Queries the delta Subcarrier or frequency value (depending on which units have been set) at the selected marker position on the WLAN EVM display.

Use the command [SENSe]:WLAN:UNIT:FREQuency to set the units.

`<x>`, which is optional, is the desired marker to query. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. Marker 0 is not valid for the DELTa queries. If `<x>` is not supplied, the command works on the currently selected marker. Valid integral values range from 1 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

`CALCulate:WLAN:EVM:MARKer<x>:DELTa:SUBCarrier?`

Returns

Returns a floating point number that represents the delta subcarrier or frequency value at the selected marker position on the WLAN EVM display.

Examples

`CALC:WLAN:EVM:MARK3:DELTa:SUBCarrier?` might return `9.375 MHz`, indicating a delta frequency value of 9.375 MHz at the Marker 3 (M3) position.

CALCulate:WLAN:EVM:MARKer<x>:DELTa:SYMBol? (Query Only)

Queries the delta time value in seconds or symbols (depending on which units have been set) at the selected marker position on the WLAN EVM display.

Use the command [SENSe]:WLAN:UNIT:TIME to set the units.

`<x>`, which is optional, is the desired marker to query. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. Marker 0 is not valid for the DELTa queries. If `<x>` is not supplied, the command works on the currently selected marker. Valid integral values range from 1 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

`CALCulate:WLAN:EVM:MARKer<x>:DELTa:SYMBol?`

Returns

Returns the delta time value at the selected marker position on the WLAN EVM display.

Examples

`CALC:WLAN:EVM:MARK3:DELTa:SYMB?` might return `4.0E-6 µs`, indicating a delta time value of 4 µs at the Marker 3 (M3) position (assuming the units have been set to Seconds).

CALCulate:WLAN:EVM:MARKer<x>:DELTa:Y? (Query Only)

Queries the vertical delta value for the points trace at the selected marker position on the WLAN EVM display.

`<x>`, which is optional, is the desired marker to query. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. Marker 0 is not valid for the DELTa queries. If `<x>` is not supplied, the command works on the currently selected marker. Valid integral values range from 1 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

`CALCulate:WLAN:EVM:MARKer<x>:DELTa:Y?`

Returns

Returns the vertical delta value for the points trace at the selected marker position.

Examples

`CALC:WLAN:EVM:MARK3:DELTa:Y?` might return `-212.91`, indicating a vertical delta value of –212.91 percent at the Marker 3 (M3) position.

CALCulate:WLAN:EVM:MARKer<x>:FREQuency

Sets or queries the frequency value of the marker on the WLAN EVM display. The units are either Frequency in Hz or Subcarrier number. To set the units, use the command [SENSe]:WLAN:UNIT:FREQuency.

Note that FREQuency is not an actual measurement of frequency, but is the subcarrier offset within the analyzed packet burst.

`<x>`, which is optional, is the desired marker to use. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If `<x>` is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Commanding a marker which does not exist results in an execution error.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

```
CALCulate:WLAN:EVM:MARKer<x>:FREQuency <NRf>
CALCulate:WLAN:EVM:MARKer<x>:FREQuency?
```

Arguments

Floating point number that indicates the desired subcarrier or frequency to which the marker should be moved. Subcarriers are traditionally numbered with both positive and negative offsets from the CF for the modulation scheme. Valid integral values range from –maxsubcarrier to maxsubcarrier, where the actual values depend on the modulation standard. Specifying a value of <NRf> outside of the range of subcarriers present in the burst selects the nearest valid subcarrier. Subcarriers can also be specified by frequency (also determined by modulation standard).

Returns

Returns a unitless number representing the subcarrier number or frequency in Hz.

Examples

`CALC:WLAN:EVM:MARK2:FREQ 2.0000` sets the position of Marker 2 (M2) to Subcarrier 2 (assuming the units have been set to Subcarrier).

`CALC:WLAN:EVM:MARK2:FREQ?` might return `2.41575E+9`, indicating that Marker 2 (M2) is positioned at 2.41575 MHz.

CALCulate:WLAN:EVM:MARKer<x>:TIME

Sets the vertical marker value of the points trace on the WLAN EVM display. The units are either Symbols or Seconds. To set/query the units, use the command [SENSe]:WLAN:UNIT:TIME.

Note that TIME is a measurement of neither absolute nor elapsed time, but symbol offset within the analyzed packet burst.

`<x>`, which is optional, is the desired marker to use. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If `<x>` is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Commanding a marker which does not exist results in an execution error.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

```
CALCulate:WLAN:EVM:MARKer<x>:TIME <NRf>
CALCulate:WLAN:EVM:MARKer<x>:TIME?
```

Arguments

Floating point number that indicates the 0 based symbol or time offset from the start of packet burst. Valid integral values range from 0 to the number of symbols in the burst. Specifying a value outside of the range of symbols present in the burst selects the nearest valid symbol.

Returns

Returns a unitless integer value in floating point format that represents the symbol associated with the position of the indicated marker, or the time offset.

Examples

`CALC:WLAN:EVM:MARK3:TIME 362.42E-6` moves Marker 3 (M3) to the `362.42 µs` position on the trace (assuming the units have been set to Seconds).

`CALC:WLAN:EVM:MARK3:TIME?` might return 5.0000 indicating that Marker 3 is located at symbol 5 (assuming units have been set to Symbols).

CALCulate:WLAN:EVM:MARKer<x>:VALue? (Query Only)

Queries the EVM value at the selected marker position on the points trace of the WLAN EVM display.

The EVM value of a marker on the average trace is queried by using the and CALCulate:WLAN:EVM:MARKer<x>:AVERage:SYMBol? queries.

`<x>`, which is optional, is the desired marker to query. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If `<x>` is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

```
CALCulate:WLAN:EVM:MARKer<x>:VALue?
```

Returns

Returns the EVM value at the selected marker position on the WLAN EVM display.

Examples

`CALC:WLAN:EVM:MARK3:VAL?` might return `11.9`, indicating an EVM value of 11.9 percent for Marker 3 (M3).

CALCulate:WLAN:FLATness:MARKer<x>:MAXimum (No Query Form)

Positions the specified marker at the maximum point on the trace in the WLAN Spectral Flatness display.

NOTE: WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

`CALCulate:WLAN:FLATness:MARKer<x>:MAXimum`

Arguments

None

Examples

`CALC:WLAN:FLAT:MARK2:MAX` positions the Marker 2 (M2) at the maximum point on the trace.

CALCulate:WLAN:FLATness:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the specified marker to the next peak higher in amplitude on the trace on the WLAN Spectral Flatness display.

NOTE: WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

`CALCulate:WLAN:FLATness:MARKer<x>:PEAK:HIGHer`

Arguments

None

Examples

`CALC:WLAN:FLAT:MARK2:PEAK:HIGH` moves Marker 2 (M2) to the next peak higher in amplitude on the trace.

CALCulate:WLAN:FLATness:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the specified marker to the next peak to the left on the trace in the WLAN Spectral Flatness display.

NOTE: *WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.*

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

CALCulate:WLAN:FLATness:MARKer<x>:PEAK:LEFT

Arguments

None

Examples

CALC:WLAN:FLAT:MARK2:PEAK:LEFT moves Marker 2 (M2) to the next peak to the left on the trace.

CALCulate:WLAN:FLATness:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the marker to the next peak lower in amplitude on the trace on the WLAN Spectral Flatness display.

NOTE: WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

CALCulate:WLAN:FLATness:MARKer<x>:PEAK:LOWer

Arguments

None

Examples

CALC:WLAN:FLAT:MARK2:PEAK:LOW moves Marker 2 (M2) to the next peak lower on the trace.

CALCulate:WLAN:FLATness:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the marker to the next peak to the right on the trace on the WLAN Spectral Flatness display.

NOTE: WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

CALCulate:WLAN:FLATness:MARKer<x>:PEAK:RIGHt

Arguments

None

Examples

`CALC:WLAN:FLAT:MARK2:PEAK:RIGH` moves Marker 2 (M2) to the next peak to the next peak on the right on the trace.

CALCulate:WLAN:FLATness:MARKer<x>:X

Sets or queries the horizontal position (units in Frequency or Subcarrier) for the selected marker on the points trace in the WLAN Spectral Flatness display.

To set the units, use the command `[SENSe]:WLAN:UNIT:FREQuency`.

NOTE: WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

```
CALCulate:WLAN:FLATness:MARKer<x>:X <NRf>
CALCulate:WLAN:FLATness:MARKer<x>:X?
```

Arguments

Floating point number that specifies the horizontal position of the marker. Range:

Start to Stop frequency (left to right edge of the horizontal axis). Using an out-of-range value causes an execution error (-222, “Data out of range”).

Examples

`CALC:WLAN:FLAT:MARK3:X 22` places Marker 3 at 22 Subcarrier on the trace (assuming the units have been set to Subcarrier).

CALCulate:WLAN:FLATness:MARKer<x>:Y? (Query Only)

Returns the value of the amplitude (vertical position) at the selected marker position in the WLAN Spectral Flatness measurement.

NOTE: WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

```
CALCulate:WLAN:FLATness:MARKer<x>:Y?
```

Returns

Floating point number that indicates the amplitude at the selected marker position in dB.

Examples

`CALC:WLAN:FLAT:MARK1:Y?` might return 33.393856E-3 indicating the amplitude of Marker 1 is 0.033 dB.

CALCulate:WLAN:MERRor:MARKer<x>:AVERage:SUBCarrier? (Query Only)

Queries the value readout at the selected marker position on the average trace on the WLAN Magnitude Error display.

To query the value readout on the symbol graph, use `CALCulate:WLAN: MERRor:MARKer<x>:AVERage:SYMBol?`

`<x>`, which is optional, is the desired marker to query. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If `<x>` is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

`CALCulate:WLAN:MERRor:MARKer<x>:AVERage:SUBCarrier?`

Returns

Returns a floating point number that represents the value readout at the selected marker position on the average trace on the WLAN Magnitude Error display subcarrier graph.

Examples

`CALC:WLAN:MERR:MARK2:AVERAGE:SUBC?` might return 16.4, indicating a value readout of 16.4 percent at the Marker 3 (M3) position on the average trace on the WLAN Magnitude Error display subcarrier graph.

CALCulate:WLAN:MERRor:MARKer<x>:AVERage:SYMBol? (Query Only)

Queries the value readout at the selected marker position on the average trace on the WLAN Magnitude Error display.

To query the value readout on the subcarrier graph, use `CALCulate:WLAN:MERRor:MARKer<x>:AVERage:SUBCarrier?`

`<x>`, which is optional, is the desired marker to query. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If `<x>` is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

```
CALCulate:WLAN:MERRor:MARKer<x>:AVERage:SYMBol?
```

Returns

Returns a floating point number that represents the value readout at the selected marker position on the average trace on the WLAN Magnitude Error display symbol graph.

Examples

`CALC:WLAN:MERR:MARK2:AVERAGE:SUBC?` might return `14.5`, indicating a Magnitude Error value of 14.5 percent at the Marker 3 (M3) position on the average trace on the WLAN Magnitude Error display.

CALCulate:WLAN:MERRor:MARKer<x>:DELTa:SUBCarrier? (Query Only)

Queries the delta Subcarrier or frequency value (depending on which units have been set) at the selected marker position on the WLAN Magnitude Error display.

Use the command [SENSe]:WLAN:UNIT:FREQuency to set the units.

<x>, which is optional, is the desired marker to query. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. Marker 0 is not valid for the DELTa queries. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 1 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

```
CALCulate:WLAN:MERRor:MARKer<x>:DELTa:SUBCarrier?
```

Returns

Returns a floating point number that represents the delta subcarrier or frequency value at the selected marker position on the WLAN Magnitude Error display.

Examples

`CAL:WLAN:MERR:MARK2:DELTA:SUBC?` might return `9.375 MHz`, indicating a delta frequency value of 9.375 MHz at the Marker 3 (M3) position.

CALCulate:WLAN:MERRor:MARKer<x>:DELTa:SYMBol? (Query Only)

Queries the delta time value in seconds or symbols (depending on which units have been set) at the selected marker position on the WLAN Magnitude Error display.

Use the command `[SENSe]:WLAN:UNIT:TIME` to set the units.

<x>, which is optional, is the desired marker to query. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. Marker 0 is not valid for the DELTa queries. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 1 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

```
CALCulate:WLAN:MERRor:MARKer<x>:DELTa:SYMBol?
```

Returns

Returns the delta time value at the selected marker position on the WLAN Magnitude Error display.

Examples

`CALC:WLAN:MERR:MARK3:DELTa:SYMB?` might return `4.0E-6 µs`, indicating a delta time value of 4 µs at the Marker 3 (M3) position (assuming the units have been set to Seconds).

CALCulate:WLAN:MERRor:MARKer<x>:DELTa:Y? (Query Only)

Queries the vertical delta value for the points trace at the selected marker position on the WLAN Magnitude Error display.

<x>, which is optional, is the desired marker to query. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. Marker 0 is not valid for the DELTa queries. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 1 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

```
CALCulate:WLAN:MERRor:MARKer<x>:DELTa:Y?
```

Returns

Returns the vertical delta value for the points trace at the selected marker position.

Examples

`CALC:WLAN:MERR:MARK3:DELTa:Y?` might return `-212.91`, indicating a vertical delta value of −212.91 percent at the Marker 3 (M3) position.

CALCulate:WLAN:MERRor:MARKer<x>:FREQuency

Sets or queries the frequency value of the marker on the WLAN Magnitude Error display. The units are either Frequency in Hz or Subcarrier number. To set the units, use the command `[SENSe]:WLAN:UNIT:FREQuency`.

Note that FREQuency is not an actual measurement of frequency, but is the subcarrier offset within the analyzed packet burst.

`<x>`, which is optional, is the desired marker to use. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If `<x>` is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Commanding a marker which does not exist results in an execution error.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

```
CALCulate:WLAN:MERRor:MARKer<x>:FREQuency <NRf>
CALCulate:WLAN:MERRor:MARKer<x>:FREQuency?
```

Arguments

Floating point number that indicates the desired subcarrier or frequency to which the marker should be moved. Subcarriers are traditionally numbered with both positive and negative offsets from the CF for the modulation scheme. Valid integral values range from –maxsubcarrier to maxsubcarrier, where the actual values depend on the modulation standard. Specifying a value of `<NRf>` outside of the range of subcarriers present in the burst selects the nearest valid subcarrier. Subcarriers can also be specified by frequency (also determined by modulation standard).

Returns

Returns a unitless number representing the subcarrier number or frequency value in Hz.

Examples

`CALC:WLAN:MERR:MARK2:FREQ 2.0000` sets the position of Marker 2 (M2) to Subcarrier 2 (assuming the units have been set to Subcarrier).

`CALC:WLAN:MERR:MARK2:FREQ?` might return `2.41575E+9`, indicating that Marker 2 (M2) is positioned at 2.41575 MHz.

CALCulate:WLAN:MERRor:MARKer<x>:TIME

Sets the time offset value at the selected marker position on the points trace on the WLAN Magnitude Error display. The units are either Symbols or Seconds. To set/query the units, use the command `[SENSe]:WLAN:UNIT:TIME`.

Note that TIME is a measurement of neither absolute nor elapsed time, but symbol offset within the analyzed packet burst.

`<x>`, which is optional, is the desired marker. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If `<x>` is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Commanding a marker which does not exist results in an execution error.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

```
CALCulate:WLAN:MERRor:MARKer<x>:TIME <NRf>
CALCulate:WLAN:MERRor:MARKer<x>:TIME?
```

Arguments

Floating point number that indicates the 0 based symbol or time offset from the start of packet burst. Valid integral values range from 0 to the number of symbols in the burst. Specifying a value outside of the range of symbols present in the burst selects the nearest valid symbol.

Returns

Returns a unitless integer value in floating point format that represents the time associated with the position of the indicated marker, or the time offset.

Examples

`CALC:WLAN:MERR:MARK3:TIME 362.42E-6` moves Marker 3 (M3) to the 362 42 µs position on the trace (assuming the units have been set to Seconds).

`CALC:WLAN:MERR:MARK3:TIME?` might return 5.0000 indicating that Marker 3 is located at symbol 5 (assuming units have been set to Symbols).

CALCulate:WLAN:MERRor:MARKer<x>:VALue? (Query Only)

Queries the value readout at the selected marker position on the points trace of the WLAN Magnitude Error display.

The value readout of a marker on the average trace is queried by using the and CALCulate:WLAN:MERRor:MARKer<x>:AVERage:SYMBol? queries.

<x>, which is optional, is the desired marker to query. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

```
CALCulate:WLAN:MERRor:MARKer<x>:VALue?
```

Returns

Returns the decoded symbol value at the selected marker position on the WLAN Magnitude Error display.

Examples

`CALC:WLAN:MERR:MARK3:VAL?` might return 11.9, indicating a value readout of 11.9 for Marker 3 (M3).

CALCulate:WLAN:PERRor:MARKer<x>:AVERage:SUBCarrier? (Query Only)

Queries the Phase Error value at the selected marker position on the average trace on the WLAN Phase Error display subcarrier graph.

To query the marker value of the points trace, use the query.

<x>, which is optional, is the desired marker to query. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

```
CALCulate:WLAN:PERRor:MARKer<x>:AVERage:SUBCarrier?
```

Returns

Returns a floating point number that represents the Phase Error value at the selected marker position on the average trace on the WLAN Phase Error display subcarrier graph.

Examples

`CALC:WLAN:PERR:MARK2:AVERAGE:SUBC?` might return 16.4, indicating a Phase Error value of 16.4 percent at the Marker 3 (M3) position on the average trace on the WLAN Phase Error display subcarrier graph.

CALCulate:WLAN:PERRor:MARKer<x>:AVERage:SYMBol? (Query Only)

Queries the Phase Error value at the selected marker position on the average trace on the WLAN Phase Error display symbol graph.

To query the marker value of the points trace, use the query.

<x>, which is optional, is the desired marker to query. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Group

Calculate commands

Syntax

```
CALCulate:WLAN:PERRor:MARKer<x>:AVERage:SYMBol?
```

Returns

Returns a floating point number that represents the Phase Error value at the selected marker position on the average trace on the WLAN Phase Error display symbol graph.

Examples

CALC:WLAN:PERR:MARK2:AVERAGE:SUBC? might return 14.5, indicating a Phase Error value of 14.5 percent at the Marker 3 (M3) position on the average trace on the WLAN Phase Error display symbol graph.

CALCulate:WLAN:PERRor:MARKer<x>:DELTa:SUBCarrier? (Query Only)

Queries the delta Subcarrier or frequency value (depending on which units have been set) at the selected marker position on the WLAN Phase Error display.

Use the command [SENSe]:WLAN:UNIT:FREQuency to set the units.

<x>, which is optional, is the desired marker to query. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. Marker 0 is not valid for the DELTa queries. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 1 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

CALCulate:WLAN:PERRor:MARKer<x>:DELTa:SUBCarrier?

Returns

Returns a floating point number that represents the delta subcarrier or frequency value at the selected marker position on the WLAN Phase Error display.

Examples

CAL:WLAN:PERR:MARK2:DELTA:SUBC? might return 9.375 MHz, indicating a delta frequency value of 9.375 MHz at the Marker 3 (M3) position.

CALCulate:WLAN:PERRor:MARKer<x>:DELTa:SYMBol? (Query Only)

Queries the delta time value in seconds or symbols (depending on which units have been set) at the selected marker position on the WLAN Phase Error display.

Use the command [SENSe]:WLAN:UNIT:TIME to set the units.

<x>, which is optional, is the desired marker to query. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. Marker 0 is not valid for the DELTa queries. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 1 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

```
CALCulate:WLAN:PERRor:MARKer<x>:DELTa:SYMBol?
```

Returns

Returns the delta time value at the selected marker position on the WLAN Phase Error display.

Examples

`CALC:WLAN:PERR:MARK3:DELTa:SYMB?` might return `4.0E-6 µs`, indicating a delta time value of 4 µs at the Marker 3 (M3) position (assuming the units have been set to Seconds).

CALCulate:WLAN:PERRor:MARKer<x>:DELTa:Y? (Query Only)

Queries the vertical delta value for the points trace at the selected marker position on the WLAN Phase Error display.

<x>, which is optional, is the desired marker to query. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. Marker 0 is not valid for the DELTa queries. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 1 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

```
CALCulate:WLAN:PERRor:MARKer<x>:DELTa:Y?
```

Returns

Returns the vertical delta value for the points trace at the selected marker position.

Examples

`CALC:WLAN:PERR:MARK3:DELTa:Y?` might return `-212.91`, indicating a vertical delta value of -212.91 percent at the Marker 3 (M3) position.

CALCulate:WLAN:PERRor:MARKer<x>:FREQuency

Sets or queries the frequency value at the selected marker position on the WLAN Phase Error display. The units are either Frequency in Hz or Subcarrier number. To set the units, use the command [SENSe]:WLAN:UNIT:FREQuency.

Note that FREQuency is not an actual measurement of frequency, but is the subcarrier offset within the analyzed packet burst.

<x>, which is optional, is the desired marker to use. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Commanding a marker which does not exist results in an execution error.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

```
CALCulate:WLAN:PERRor:MARKer<x>:FREQuency <NRf>
CALCulate:WLAN:PERRor:MARKer<x>:FREQuency?
```

Arguments

Floating point number that indicates the desired subcarrier or frequency to which the marker should be moved. Subcarriers are traditionally numbered with both positive and negative offsets from the CF for the modulation scheme. Valid integral values range from –maxsubcarrier to maxsubcarrier, where the actual values depend on the modulation standard. Specifying a value of <NRf> outside of the range of subcarriers present in the burst selects the nearest valid subcarrier. Subcarriers can also be specified by frequency (also determined by modulation standard).

Returns

Returns a unitless number representing the subcarrier number or frequency value in Hz.

Examples

`CALC:WLAN:PERR:MARK2:FREQ 2.0000` sets the position of Marker 2 (M2) to Subcarrier 2 (assuming the units have been set to Subcarrier).

`CALC:WLAN:PERR:MARK2:FREQ?` might return `2.41575E+9`, indicating that Marker 2 (M2) is positioned at 2.41575 MHz.

CALCulate:WLAN:PERRor:MARKer<x>:TIME

Sets the time value (offset) at the selected marker position on the points trace on the WLAN Phase Error display. The units are either Symbols or Seconds. To set/query the units, use the command `[SENSe]:WLAN:UNIT:TIME`.

Note that TIME is a measurement of neither absolute nor elapsed time, but symbol offset within the analyzed packet burst.

`<x>`, which is optional, is the desired marker to use. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If `<x>` is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Commanding a marker which does not exist results in an execution error.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

```
CALCulate:WLAN:PERRor:MARKer<x>:TIME <NRf>
CALCulate:WLAN:PERRor:MARKer<x>:TIME?
```

Arguments

Floating point number that indicates the 0 based symbol or time offset from the start of packet burst. Valid integral values range from 0 to the number of symbols in the burst. Specifying a value outside of the range of symbols present in the burst selects the nearest valid symbol.

Returns

Returns a unitless integer value in floating point format that represents the symbol associated with the position of the indicated marker, or the time offset.

Examples

`CALC:WLAN:PERR:MARK3:TIME 362.42E-6` moves Marker 3 (M3) to the 362.42 µs position on the trace (assuming the units have been set to Seconds).

`CALC:WLAN:PERR:MARK3:TIME?` might return `5.0000` indicating that Marker 3 is located at symbol 5 (assuming units have been set to Symbols).

CALCulate:WLAN:PERRor:MARKer<x>:VALue? (Query Only)

Queries the Phase Error value at the selected marker position on the points trace of the WLAN Phase Error display.

The Phase Error value of a marker on the average trace is queried by using the and `CALCulate:WLAN:PERRor:MARKer<x>:AVERage:SYMBol?` queries.

`<x>`, which is optional, is the desired marker to query. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If `<x>` is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

`CALCulate:WLAN:PERRor:MARKer<x>:VALue?`

Returns

Returns the Phase Error value at the selected marker position on the WLAN Phase Error display.

Examples

`CALC:WLAN:PERR:MARK3:VAL?` might return `11.9`, indicating a Phase Error value of 11.9 percent for Marker 3 (M3).

CALCulate:WLAN:PVTime:MARKer<x>:DELTa:X? (Query Only)

Returns the delta time value at the selected marker position in the WLAN Power vs. Time display.

`<x>` is the desired marker to query. Marker 0 is not valid for the DELTa queries. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. Valid integral values range from 1 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

```
CALCulate:WLAN:PVTime:MARKer<x>:DELTa:X?
```

Returns

Floating point number that represents the delta time value at the selected marker position.

Examples

`CALC:WLAN:PVT:MARK2:DELT:X?` might return `1.28E-6` indicating that the delta time value for Marker 2 (M2) is 1.28 µs.

CALCulate:WLAN:PVTime:MARKer<x>:DELTa:Y? (Query Only)

Queries the vertical delta value for the points trace at the selected marker position on the WLAN Power vs. Time display, using the currently selected Amplitude unit. Set/query the amplitude unit with the command SENSe:POWer:UNITs.

`<x>` is the desired marker to query. Marker 0 is not valid for the DELTa queries. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. Valid integral values range from 1 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

```
CALCulate:WLAN:PVTime:MARKer<x>:DELTa:Y?
```

Returns

Returns the vertical delta value for the points trace in the current Amplitude units at the selected marker position

.

Examples

`CALC:WLAN:PVT:MARK3:DELT:Y?` might return `-12.91`, indicating a vertical delta value of −12.91 dB at the Marker 3 (M3) position.

CALCulate:WLAN:PVTime:MARKer<x>:MAXimum (No Query Form)

Positions the specified marker at the maximum point on the trace in the WLAN Power vs. Time display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

```
CALCulate:WLAN:PVTime:MARKer<x>:MAXimum
```

Arguments

None

Examples

`CALC:WLAN:PVT:MARK2:MAX` positions the Marker 2 (M2) at the maximum point on the trace.

CALCulate:WLAN:PVTime:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the specified marker to the next peak higher in amplitude on the trace on the WLAN Power vs. Time display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

`CALCulate:WLAN:PVTime:MARKer<x>:PEAK:HIGHer`

Arguments

None

Examples

`CALC:WLAN:PVT:MARK2:PEAK:HIGHer` moves Marker 2 (M2) to the next peak higher in amplitude on the trace.

CALCulate:WLAN:PVTime:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the specified marker to the next peak on the left on the trace in the WLAN Power vs. Time display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

`CALCulate:WLAN:PVTime:MARKer<x>:PEAK:LEFT`

Arguments

None

Examples

`CALC:WLAN:PVT:MARK2:PEAK:LEFT` moves Marker 2 (M2) to the next peak on the left on the trace in the WLAN Power vs. Time display.

CALCulate:WLAN:PVTime:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the marker to the next peak lower in amplitude on the trace on the WLAN Power vs. Time display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

`CALCulate:WLAN:PVTime:MARKer<x>:PEAK:LOWer`

Arguments

None

Examples

`CALC:WLAN:PVT:MARK2:PEAK:LOW` moves Marker 2 (M2) to the next peak lower on the trace.

CALCulate:WLAN:PVTime:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the marker to the next peak to the right on the trace on the WLAN Power vs. Time display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

`CALCulate:WLAN:PVTime:MARKer<x>:PEAK:RIGHt`

Arguments

None

Examples

`CALC:WLAN:PVT:MARK2:PEAK:RIGH` moves Marker 2 (M2) to the next peak on the right on the trace.

CALCulate:WLAN:PVTime:MARKer<x>:X

Sets or queries the value of the horizontal position (in seconds) for the selected marker on the points trace in the WLAN Power vs. Time display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

```
CALCulate:WLAN:PVTime:MARKer<x>:X <NRf>
CALCulate:WLAN:PVTime:MARKer<x>:X?
```

Arguments

Floating point number that specifies the value of the horizontal position of the marker. Range: Start to Stop time (left to right edge of the horizontal axis). Using an out-of-range value causes an execution error (-222, "Data out of range").

Examples

`CALC:WLAN:PVT:MARK3:X 1E-6` places Marker 3 at 1 µs on the trace.

CALCulate:WLAN:PVTime:MARKer<x>:Y? (Query Only)

Returns the amplitude (vertical position) at the selected marker position in the WLAN Power vs. Time display, using the currently selected Amplitude units. You can set/query the Amplitude units with the command `SENSe:POWer:UNITs`.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

```
CALCulate:WLAN:PVTime:MARKer<x>:Y?
```

Returns

Floating point number that indicates the amplitude at the selected marker position in the currently selected Amplitude units.

Examples

`CALC:WLAN:PVT:MARK1:Y?` might return `33.393856E-3` indicating the amplitude at the Marker 1 position is 0.033 dBm.

CALCulate:WLAN:TABLe:MARKer<x>:FREQuency

Assigns or queries the frequency value of the marker in the WLAN Symbol Table. The units are either Frequency in Hz or Subcarrier number. To set the units, use the command [SENSe]:WLAN:UNIT:FREQuency.

Note that FREQuency is not an actual measurement of frequency, but is the subcarrier offset within the analyzed packet burst.

<x>, which is optional, is the desired marker to assign. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commandsCALCulate:WLAN:AY:CONSte:MARKer<x>:DELTa:X:[:TIME]? (Query Only)

Syntax

```
CALCulate:WLAN:TABLe:MARKer<x>:FREQuency <NRf>
CALCulate:WLAN:TABLe:MARKer<x>:FREQuency?
```

Arguments

Floating point number that indicates the desired subcarrier or frequency to which the marker should be assigned. Subcarriers are traditionally numbered with both positive and negative offsets from the CF for the modulation scheme.Valid integral values range from –maxsubcarrier to maxsubcarrier, where the actual values depend on the modulation standard. Specifying a value of `<NRf>` outside of the range of subcarriers present in the burst selects the nearest valid subcarrier. Subcarriers can also be specified by frequency (also determined by modulation standard).

Returns

Returns a unitless number representing the subcarrier number or frequency in Hz.

Examples

`CALC:WLAN:TABLE:MARK3:FREQ 2.0000` assigns Subcarrier 2 to Marker 3 (M3) (assuming the units have been set to Subcarrier).

`CALC:WLAN:TABLEM:MARK2:FREQ?` might return `2.00000E+9`, indicating that Marker 2 (M2) is assigned 2.00 MHz.

CALCulate:WLAN:TABLe:MARKer<x>:TIME

Sets or queries the vertical value at the selected marker position on the WLAN Symbol Table. The units are either Symbols or Seconds. To set/query the units, use the command `[SENSe]:WLAN:UNIT:TIME`.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

```
CALCulate:WLAN:TABLe:MARKer<x>:TIME <NRf>
CALCulate:WLAN:TABLe:MARKer<x>:TIME?
```

Arguments

Floating point number that indicates the 0 based symbol or time offset from the start of packet burst. Valid integral values range from 0 to the number of symbols in the burst. Specifying a value outside of the range of symbols present in the burst selects the nearest valid symbol.

Returns

Returns a unitless integer value in floating point format that represents the symbol associated with the indicated marker, or the time offset.

Examples

`CALC:WLAN:TABLE:MARK3:TIME 362.42E-6` assigns `362.42 µs` to Marker 3 (M3) (assuming the units have been set to Seconds).

`CALC:WLAN:TABLE:MARK3:TIME?` might return `5.0000` indicating that symbol 5 is assigned to Marker 3.

CALCulate:WLAN:TABLe:MARKer<x>:VALue? (Query Only)

Queries the decoded symbol value at the selected marker position on the WLAN Symbol Table display.

`<x>`, which is optional, is the desired marker to query. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If `<x>` is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

`CALCulate:WLAN:TABLe:MARKer<x>:VALue?`

Returns

Returns the decoded symbol value at the selected marker position on the WLAN Symbol Table display.

Examples

`CALC:WLAN:MERR:MARK3:VAL?` might return `11.9`, indicating a decoded symbol value of 11.9 for Marker 3 (M3).

Calibration commands

CALibration:ABORt (No query form)

Aborts any actions related to the alignments in progress.

Conditions

Measurement views: All

Group

Calibration commands

Syntax

`CALibration:ABORt`

Arguments

None

Examples

`CALibration:ABORt` aborts any actions related to the alignments in progress.

CALibration:AUTO

Sets or queries whether or not to run alignments automatically.

Conditions

Measurement views: All

Group

Calibration commands

Syntax

```
CALibration:AUTO { OFF | ON | 0 | 1 }
CALibration:AUTO?
```

Arguments

`OFF` or 0 runs alignments on user request.

Use the `*CAL` command to perform alignments.

`ON` or 1 runs alignments as needed without user intervention.

You have to restart measurement if interrupted.

Examples

`CALibration:AUTO ON` runs alignments automatically as needed.

CALibration:CORRection:EXTernal:EDIT<x>:LABel

Sets or queries the name of the external loss table.

The parameter <x> = 1 to 3 represent the External Loss Table 1 to 3, respectively.

NOTE: This command has been replaced with `INPut:CORRection:EXTernal:EDIT<x>:LABel`.

Conditions

Measurement views: All

Group

Calibration commands

Syntax

```
CALibration:CORRection:EXTernal:EDIT<x>:LABel <***Set NAME variable****>
CALibration:CORRection:EXTernal:EDIT<x>:LABel?
```

Arguments

<***Set NAME variable****>`::=<string>` specifies the name of the external loss table.

Examples

`CALibration:CORRection:EXTernal:EDIT<x>:LABel? 'Sample Table 1'` names the External Loss Table 1 "Sample Table 1".

CALibration:CORRection:EXTernal:EDIT<x>:NEW (No query form)

Creates a new external loss table.

The parameter <x> = 1 to 3 represent the External Loss Table 1 to 3, respectively.

NOTE: This command has been replaced with `INPut:CORRection:EXTernal:EDIT<x>:NEW`.

Conditions

Measurement views: All

Group

Calibration commands

Syntax

```
CALibration:CORRection:EXTernal:EDIT<x>:NEW
<freq(1)>,<loss(1)>,<freq(2)>,<loss(2)>,...,<freq(n)>,<loss(n)>
```

Arguments

`<freq(n)>,<loss(n)>` specifies a pair of frequency (`<NR3>` in Hz) and loss (`<NR3>` in dB) in the external loss table.

The setting range is:

- RSA5103B – 0 to 3 GHz
- RSA5115B – 0 to 15 GHz
- RSA5126B – 0 to 26.5 GHz

Loss: -50 to +80 dB. (A negative value means a gain.)

Examples

`CALibration:CORRection:EXTernal:EDIT<x>:NEW 1.0E+9,2.2, 1.5E+9,2.3` creates the External Loss Table 1 specifying the loss of 2.2 dB and 2.3 dB at the frequency of 1 GHz and 1.5 GHz, respectively.

CALibration:CORRection:EXTernal:EDIT<x>:STATe

Determines whether to enable or disable the external loss table.

The parameter <x> = 1 to 3 represent the External Loss Table 1 to 3, respectively.

NOTE: This command has been replaced with `INPut:CORRection:EXTernal:EDIT<x>:STATe`.

Conditions

Measurement views: All

Group

Calibration commands

Syntax

```
CALibration:CORRection:EXTernal:EDIT<x>:STATe { OFF | ON | 0 | 1 }
CALibration:CORRection:EXTernal:EDIT<x>:STATe?
```

Arguments

`OFF` or 0 disables the external loss table.

`ON` or 1 enables the external loss table.

NOTE: You can enable one or more tables at the same time.

Examples

CALibration:CORRection:EXTernal:EDIT<x>:STATe? ON enables the External Loss Table 3.

CALibration:CORRection:EXTernal:GAIN[:MAGNitude]

Sets or queries the external gain value. It can be enabled or disabled using the INPut:CORRection:EXTernal:GAIN:STATe command.

NOTE: This command has been replaced with INPut:CORRection:EXTernal:GAIN[:MAGNitude].

Conditions

Measurement views: All

Group

Calibration commands

Syntax

```
CALibration:CORRection:EXTernal:GAIN[:MAGNitude] <value>
CALibration:CORRection:EXTernal:GAIN[:MAGNitude]?
```

Arguments

<value>::=<NRf> specifies the external gain value.

Range: -50 to +30 dB.

Examples

CALibration:CORRection:EXTernal:GAIN[:MAGNitude]? -10 sets the external gain to -10 dB.

CALibration:CORRection:EXTernal:GAIN:STATe

Determines whether to enable or disable the external gain value.

NOTE: This command has been replaced with INPut:CORRection:EXTernal:GAIN[:MAGNitude].

Conditions

Measurement views: All

Group

Calibration commands

Syntax

```
CALibration:CORRection:EXTernal:GAIN:STATe { OFF | ON | 0 | 1 }
CALibration:CORRection:EXTernal:GAIN:STATe?
```

Related Commands

INPut:CORRection:EXTernal:GAIN[:MAGNitude] (on page 660)

Arguments

OFF or 0 disables the external gain value.

ON or 1 enables the external gain value.

Examples

`CALibration:CORRection:EXTernal:GAIN:STATe ON` enables the external gain value.

CALibration:CORRection:EXTernal:PROBe:CONNect? (Query only)

Queries whether the external probe is connected to the analyzer or not.

Conditions

Measurement views: All

Group

Calibration commands

Syntax

`CALibration:CORRection:EXTernal:PROBe:CONNect?`

Arguments

None

Returns

`{ 0 | 1 }`

`0` indicates that the external probe is not connected to the analyzer.

`1` indicates that the external probe is connected to the analyzer.

Examples

`CALibration:CORRection:EXTernal:PROBe:CONNect?` might return ON, indicating that the external probe is connected to the analyzer.

CALibration:CORRection:EXTernal:PROBe[:MAGNitude]? (Query only)

Queries the external probe attenuation.

Conditions

Measurement views: All

Group

Calibration commands

Syntax

`CALibration:CORRection:EXTernal:PROBe[:MAGNitude]?`

Arguments

None

Returns

`<attenuation>::=<NRf>` The probe attenuation value in dB.

Examples

CALibration:CORRection:EXTernal:PROBe[:MAGNitude]? might return 10, indicating that the probe attenuation is 10 dB.

CALibration:CORRection:EXTernal:PROBe:STATe

Determines whether or not to correct data for the external probe attenuation.

Conditions

Measurement views: All

Group

Calibration commands

Syntax

```
CALibration:CORRection:EXTernal:PROBe:STATe { OFF | ON | 0 | 1 }
CALibration:CORRection:EXTernal:PROBe:STATe?
```

Arguments

OFF or 0 does not correct data for the external probe attenuation.

ON or 1 corrects data for the external probe attenuation.

Examples

CALibration:CORRection:EXTernal:PROBe:STATe ON corrects data for the external probe attenuation.

CALibration:CORRection:EXTernal:TYPE

Sets or queries the data type to apply the external loss table corrections.

NOTE: This command has replaced with INPut:CORRection:EXTernal:TYPE

Conditions

Measurement views: All

Group

Calibration commands

Syntax

```
CALibration:CORRection:EXTernal:TYPE { TRACe | DATA }
CALibration:CORRection:EXTernal:TYPE
```

Related Commands

INPut:CORRection:EXTernal:GAIN:STATe (on page 1484)

Arguments

TRACe selects traces in the Spectrum, Spectrogram, Spurious, and Amplitude versus Time views. Selecting TRACe disables the CALibration:CORRection:EXTernal:GAIN:STATe command.

DATA selects all acquired data. Selecting DATA enables the CALibration:CORRection:EXTernal:GAIN:STATe command.

Examples

`CALibration:CORRection:EXTernal:TYPE TRACe` selects traces to apply the external loss table corrections.

CALibration:TXGain:ABORt (No query form)

Cancels a calibration.

Conditions

Measurement view: Transmission Gain, Perform Calibration window

Group

Calibration commands

Syntax

`CALibration:TXGain:ABORt`

Arguments

None

Examples

`CALibration:TXGain:ABORt` cancels a calibration that is in process.

CALibration:TXGain[:ALL] (No query form)

Performs a calibration using the current settings.

Conditions

Measurement view: Transmission Gain, Perform Calibration window

Group

Calibration commands

Syntax

`CALibration:TXGain[:ALL]`

Arguments

None

Examples

`CALibration:TXGain[:ALL]` performs a calibration using the current settings.

CALibration:TXGain:FINish? (Query only)

Queries whether or not the calibration is completed.

Conditions

Measurement view: Transmission Gain, Perform Calibration window

Group

Calibration commands

Syntax

`CALibration:TXGain:FINish?`

Returns

`0` indicates calibration is in progress.

`1` indicates calibration is completed.

Examples

`CALibration:TXGain:FINish?` might return 1, indicating that the calibration is completed.

CALibration:TXGain:STATe? (Query only)

Queries the calibration state for the Transmission Gain measurement.

Conditions

Measurement view: Transmission Gain

Group

Calibration commands

Syntax

`CALibration:TXGain:STATe?`

Returns

`CALIBRATED` means that the unit has been calibrated and is making measurements at the same frequencies at which it was calibrated.

`CALIBRATED?` means that the unit has been calibrated, but one or more of the operating conditions has changed. Operating conditions are Tracking Generator Output Power, Reference Level, Preamp On/Off, RF Attenuation, or Temperature (>5 °C from the calibrated temperature).

`INTERPOLATED` means that the unit has been calibrated, but is making measurements at frequencies other than the calibration points and is interpolating the calibration between points.

Examples

`CALibration:TXGain:STATe?` might return CALIBRATED, indicating that the measurement has been calibrated with the current settings.

*CLS (No Query Form)

Clears the analyzer status data structures. Refer to Section 3, *Status and Events*, for the register information.

The *CLS command clears the following

- the Event Queue
- the Standard Event Status Register (SESR)
- the Status Byte Register (except the MAV bit; see below)

If the *CLS command immediately follows an <EOI>, the Output Queue and MAV bit (Status Byte Register bit 4) are also cleared. MAV indicates information is in the output queue. The device clear (DCL) GPIB control message will clear the output queue and thus MAV. *CLS does not clear the output queue or MAV. (A complete discussion of these registers and bits, and of event handling in general is described in the *Status and Events* section)

*CLS can suppress a Service Request that is to be generated by an *OPC. This will happen if a hardcopy output or single sequence acquisition operation is still being processed when the *CLS command is executed.

Conditions

Measurement views: All

Group

IEEE common commands

Syntax

```
*CLS
```

Related Commands

*ESE (on page 1059)

*ESR? (on page 1059)

*SRE (on page 2482)

*STB? (on page 2531)

Arguments

None

Examples

`*CLS` clears the analyzer status data structures.

Display commands

DISPlay:ACPower:MARKer:SHOW:STATe

Determines whether to show or hide the readout for the selected marker in the Channel power and ACPR view.

Conditions

Measurement views: Channel power and ACPR

Group

Display commands

Syntax

```
DISPlay:ACPower:MARKer:SHOW:STATe { OFF | ON | 0 | 1 }
DISPlay:ACPower:MARKer:SHOW:STATe?
```

Arguments

`OFF` or 0 hides the readout for the selected marker in the graph.

`ON` or 1 shows the readout for the selected marker in the graph.

Examples

`DISPLAY:ACPOWER:MARKER:SHOW:STATE ON` shows the readout for the selected marker in the graph.

DISPlay:ACPower:PLEVel:SHOW:STATe

Determines whether to show or hide the power levels in the Channel power and ACPR view.

Conditions

Measurement views: Channel power and ACPR

Group

Display commands

Syntax

```
DISPlay:ACPower:PLEVel:SHOW:STATe { OFF | ON | 0 | 1 }
DISPlay:ACPower:PLEVel:SHOW:STATe?
```

Arguments

`OFF` or 0 hides the power levels in the graph.

`ON` or 1 shows the power levels in the graph.

Examples

`DISPLAY:ACPOWER:PLEVEL:SHOW:STATE ON` shows the power levels in the graph.

DISPlay:ACPower:RESet:SCALe (No Query Form)

Resets the horizontal and vertical scale to the default values described below in the Channel power and ACPR view.

Vertical offset = Reference level, Vertical scale = 100 dB,

Horizontal offset = Center frequency, and Horizontal scale = Default span

Conditions

Measurement views: Channel power and ACPR

Group

Display commands

Syntax

```
DISPlay:ACPower:RESet:SCALe
```

Arguments

None

Examples

`DISPLAY:ACPOWER:RESET:SCALE` resets the horizontal and vertical scale to the default values.

DISPlay:ACPower:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions

Measurement views: Channel power and ACPR

Group

Display commands

Syntax

```
DISPlay:ACPower:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:ACPower:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or 0 hides the graticule grid.

`ON` or 1 shows the graticule grid.

Examples

`DISPLAY:ACPOWER:WINDOW:TRACE:GRATICULE:GRID:STATE ON` shows the graticule grid on the screen.

DISPlay:ACPower:X[:SCALe]

Sets or queries the horizontal range of the Channel power and ACPR graph.

Conditions

Measurement views: Channel power and ACPR

Group

Display commands

Syntax

```
DISPlay:ACPower:X[:SCALe] <value>
DISPlay:ACPower:X[:SCALe]?
```

Related Commands

DISPlay:ACPower:X[:SCALe]:OFFSet (on page 668)

Arguments

`<value>::=<NRf>` specifies the horizontal range.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples

`DISPLAY:ACPOWER:X:SCALE 10MHz` sets the horizontal range to 10 MHz.

DISPlay:ACPower:X[:SCALe]:AUTO (No Query Form)

Rescales the horizontal axis automatically to fit the waveform to the screen in the Channel power and ACPR view.

Conditions

Measurement views: Channel power and ACPR

Group

Display commands

Syntax

```
DISPlay:ACPower:X[:SCALe]:AUTO
```

Arguments

None

Examples

`DISPLAY:ACPOWER:X:SCALE:AUTO` rescales the horizontal scale automatically to fit the waveform to the screen.

DISPlay:ACPower:X[:SCALe]:OFFSet

Sets or queries the minimum horizontal value (left edge) of the Channel power and ACPR graph.

Conditions

Measurement views: Channel power and ACPR

Group

Display commands

Syntax

```
DISPlay:ACPower:X[:SCALe]:OFFSet <value>
DISPlay:ACPower:X[:SCALe]:OFFSet?
```

Related Commands

DISPlay:ACPower:X[:SCALe] (on page 667)

Arguments

`<value>::=<NRf>` specifies the minimum horizontal value.

Range: [(center frequency) - (X scale) × 0.9] to [(center frequency) + (X scale) × 0.9]

Examples

`DISPlay:ACPower:X:SCALe:OFFSet 1.45GHz` sets the minimum horizontal value to 1.45 GHz in the Channel power and ACPR graph.

DISPlay:ACPower:Y[:SCALe]

Sets or queries the vertical range of the channel power and ACPR graph.

Conditions

Measurement views: Channel power and ACPR

Group

Display commands

Syntax

```
DISPlay:ACPower:Y[:SCALe] <value>
DISPlay:ACPower:Y[:SCALe]?
```

Related Commands

DISPlay:ACPower:Y[:SCALe]:OFFSet (on page 669)

Arguments

<value>::=<NRf> specifies the vertical range. Range: 0.1 to 200 dB.

Examples

DISPLAY:ACPOWER:Y:SCALE 100 sets the vertical range to 100 dB in the Channel power and ACPR graph.

DISPlay:ACPower:Y[:SCALe]:AUTO (No Query Form)

Rescales the vertical axis automatically to fit the waveform to the screen in the Channel power and ACPR view.

Conditions

Measurement views: Channel power and ACPR

Group

Display commands

Syntax

```
DISPlay:ACPower:Y[:SCALe]:AUTO
```

Arguments

None

Examples

DISPLAY:ACPOWER:Y:SCALE:AUTO rescales the vertical scale automatically to fit the waveform to the screen.

DISPlay:ACPower:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (the value at the top edge of the vertical axis) in the Channel power and ACPR graph.

Conditions

Measurement views: Channel power and ACPR

Group

Display commands

Syntax

```
DISPlay:ACPower:Y[:SCALe]:OFFSet <value>
DISPlay:ACPower:Y[:SCALe]:OFFSet?
```

Related Commands

DISPlay:ACPower:Y[:SCALe] (on page 668)

Arguments

<value>::=<NRf> specifies the vertical offset. Range: -170 to +50 dBm.

Examples

DISPLAY:ACPOWER:Y:SCALE:OFFSET -12.5 sets the vertical offset to −12.5 dBm in the Channel power and ACPR graph.

DISPlay:ADEMod:MEASview:DELete (No Query Form)

Deletes the measurement view in the general purpose analog demodulation measurements.

Conditions

Measurement views: General purpose analog demodulation

Group

Display commands

Syntax

```
DISPlay:ADEMod:MEASview:DELete { AM | FM | PM }
```

Arguments

The following table lists the arguments.

Analog demodulation measurement views

Argument	View
AM	AM measurement (modulation factor versus time)
FM	FM measurement (frequency deviation versus time)
PM	PM measurement (phase deviation versus time)

If you attempt to delete a view that is not displayed on screen, the error (-200, "Execution error; Measurement is not running") will be returned.

Examples

`DISPLAY:ADEMOD:MEASVIEW:DELETEAM` deletes the AM measurement view.

DISPlay:ADEMod:MEASview:NEW (No Query Form)

Displays a new measurement view in the general purpose analog demodulation measurements.

Conditions

Measurement views: General purpose analog demodulation

Group

Display commands

Syntax

```
DISPlay:ADEMod:MEASview:NEW { AM | FM | PM }
```

Arguments

The following table lists the arguments.

Analog demodulation measurement views

Argument	View
AM	AM measurement (modulation factor versus time)

Argument	View
FM	FM measurement (frequency deviation versus time)
PM	PM measurement (phase deviation versus time)

If you attempt to open a view that is currently displayed on screen, the error (-200, "Execution error; Measurement is already running") will be returned.

Examples

`DISPLAY:ADEMOD:MEASVIEW:NEWAM` creates the AM measurement view.

DISPlay:ADEMod:MEASview:SELect

Selects a measurement view on the screen in the general purpose analog demodulation measurements. The query command returns the currently selected view.

Selecting a measurement optimizes it. Other measurements may be optimized as a side effect. Refer to the `DISPlay:WINDow:OPTimized:MEASurement?` query.

Conditions

Measurement views: General purpose analog demodulation

Group

Display commands

Syntax

```
DISPlay:ADEMod:MEASview:SELect { AM | FM | PM }
DISPlay:ADEMod:MEASview:SELect?
```

Arguments

The following table lists the arguments.

Analog demodulation measurement views

Argument	View
AM	AM measurement (modulation factor versus time)
FM	FM measurement (frequency deviation versus time)
PM	PM measurement (phase deviation versus time)

If you attempt to select a view that is not displayed on screen, the error (-200, "Execution error; Measurement not running") will be returned.

Examples

`DISPLAY:ADEMOD:MEASVIEW:SELECT AM` selects the AM measurement view.

DISPlay:{AM|FM|PM}:MARKer:SHOW:STATe

Determines whether to show or hide the marker readout in the AM/FM/PM measurement view.

Conditions

Measurement views: General purpose analog demodulation

Group

Display commands

Syntax

```
DISPlay:{AM|FM|PM}:MARKer:SHOW:STATe { OFF | ON | 0 | 1 }
DISPlay:{AM|FM|PM}:MARKer:SHOW:STATe?
```

Arguments

`OFF` or 0 hides the marker readout on the screen.

`ON` or 1 shows the marker readout on the screen.

Examples

`DISPLAY:AM:MARKER:SHOW:STATE ON` shows the marker readout in the AM measurement view.

DISPlay:{AM|FM|PM}:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions

Measurement views: General purpose analog demodulation

Group

Display commands

Syntax

```
DISPlay:{AM|FM|PM}:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:{AM|FM|PM}:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or 0 hides the graticule grid.

`ON` or 1 shows the graticule grid.

Examples

`DISPLAY:AM:WINDOW:TRACE:GRATICULE:GRID:STATE ON` shows the graticule grid on the AM measurement view.

DISPlay:{AM|FM|PM}:X:RSCale (No Query Form)

Rescales the horizontal axis automatically to fit the waveform to the screen in the AM/FM/PM measurement display.

Conditions

Measurement views: General purpose analog demodulation

Group

Display commands

Syntax

```
DISPlay:{AM|FM|PM}:X:RSCale
```

Arguments

None

Examples

`DISPLAY:AM:X:RSCALE` rescales the horizontal scale automatically to fit the waveform to the screen.

DISPlay:{AM|FM|PM}:X[:SCALe]:AUTO (No Query Form)

Sets the horizontal scale automatically to fit the waveform to the screen in the AM/FM/PM measurement display.

Conditions

Measurement views: General purpose analog demodulation

Group

Display commands

Syntax

```
DISPlay:{AM|FM|PM}:X[:SCALe]:AUTO
```

Arguments

None

Examples

`DISPLAY:AM:X:SCALE:AUTO` sets the horizontal scale automatically to fit the waveform to the screen in the AM/FM/PM measurement display.

DISPlay:{AM|FM|PM}:X[:SCALe]:FULL

Sets or queries the horizontal scale (full-scale time) of the AM/FM/PM measurement graph.

Conditions

Measurement views: General purpose analog demodulation

Group

Display commands

Syntax

```
DISPlay:{AM|FM|PM}:X[:SCALe]:FULL <value>
DISPlay:{AM|FM|PM}:X[:SCALe]:FULL?
```

Arguments

`<value>::=<NRf>` specifies the horizontal scale in full-scale time.

Range: 0 to 1027 s.

Examples

`DISPlay:AM:X:SCALe:FULL 35us` sets the horizontal scale to 35 µs in the AM measurement.

DISPlay:{AM|FM|PM}:X[:SCALe]:OFFSet

Sets or queries the minimum horizontal value (left edge) of the AM/FM/PM measurement graph.

Conditions

Measurement views: General purpose analog demodulation

Group

Display commands

Syntax

```
DISPlay:{AM|FM|PM}:X[:SCALe]:OFFSet <value>
DISPlay:{AM|FM|PM}:X[:SCALe]:OFFSet?
```

Arguments

`<value>::=<NRf>` specifies the minimum horizontal value.

Range: −0.9×(horizontal scale) to +0.9×(horizontal scale)

Examples

`DISPLAY:AM:X:SCALE:OFFSET 20.075us` sets the minimum horizontal value to 20.075 µs.

DISPlay:{AM|FM|PM}:Y:RSCale (No Query Form)

Rescales the vertical axis automatically to fit the waveform to the screen in the AM/FM/PM measurement display.

Conditions

Measurement views: General purpose analog demodulation

Group

Display commands

Syntax

```
DISPlay:{AM|FM|PM}:Y:RSCale
```

Arguments

None

Examples

`DISPLAY:AM:Y:RSCALE` rescales the vertical scale automatically to fit the waveform to the screen in the AM measurement.

DISPlay:{AM|FM|PM}:Y[:SCALe]

Sets or queries the vertical range of the AM/FM/PM measurement graph.

Conditions

Measurement views: General purpose analog demodulation

Group

Display commands

Syntax

```
DISPlay:{AM|FM|PM}:Y[:SCALe] <value>
DISPlay:{AM|FM|PM}:Y[:SCALe]?
```

Arguments

`<value>::=<NRf>` specifies the vertical range.

The setting range depends on measurements as shown in the following table.

Setting range

Measurement	Setting range
AM	1% to 500%
FM	10 Hz to 120 MHz
PM	10° to 400°

Examples

`DISPLAY:AM:Y:SCALE 75` sets the vertical range to 75% in the AM measurement graph.

DISPlay:{AM|FM|PM}:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (center point of the vertical axis) in the AM/FM/PM measurement graph.

Conditions

Measurement views: General purpose analog demodulation

Group

Display commands

Syntax

```
DISPlay:{AM|FM|PM}:Y[:SCALe]:OFFSet <value>
DISPlay:{AM|FM|PM}:Y[:SCALe]:OFFSet?
```

Arguments

`<value>::=<NRf>` specifies the vertical offset.

The setting range depends on measurements as shown in the following table.

Setting range

Measurement	Setting range
AM	−500% to +500%
FM	–(maxVertScale * (1 + 1.1) * 0.5) to +(maxVertScale * (1 + 1.1) * 0.5)
PM	−180° to +180°

Examples

`DISPLAY:AM:Y:SCALE:OFFSET -12.5` sets the vertical offset to −12.5% in the AM measurement graph.

DISPlay:ANTenna:AZIMuth:SHOW:STATe

The command checks the Show Compass box in the UI to display the compass. The query form returns whether or not the Show Compass box is checked.

Group

Display commands

Syntax

```
DISPlay:ANTenna:AZIMuth:SHOW:STATe { 1 | 0 | ON | OFF }
DISPlay:ANTenna:AZIMuth:SHOW:STATe?
```

Arguments

1 or ON sets compass to show.

0 or OFF sets compass to hide.

Returns

1 or ON means compass is set to show.

0 or OFF means compass is set to hide.

DISPlay:AUDio:MEASview:DELete (No Query Form)

Deletes the specified audio analysis view.

Conditions

Measurement views: Audio Spectrum, Audio Summary

Group

Display commands

Syntax

```
DISPlay:AUDio:MEASview:DELete { SPECtrum | SUMMary }
```

Arguments

`SPECtrum` deletes the Audio Spectrum view.

`SUMMary` deletes the Audio Summary view.

If you attempt to delete a view that is not displayed on screen, the error (-200, "Execution error; Measurement not running") will be returned.

Examples

`DISPLAY:AUDIO:MEASVIEW:DELETE SUMMary` deletes the Audio Summary view.

DISPlay:AUDio:MEASview:NEW (No Query Form)

Displays a new audio analysis view.

Conditions

Measurement views: Audio Spectrum, Audio Summary

Group

Display commands

```
Syntax
DISPlay:AUDio:MEASview:NEW { SPECtrum | SUMMary }
```

Arguments

`SPECtrum` creates a new Audio Spectrum view.

`SUMMary` creates a new Audio Summary view.

If you attempt to open a view that is currently displayed on screen, the error (-200, "Execution error; Measurement is already running") will be returned.

Examples

`DISPLAY:AUDIO:MEASVIEW:NEW SUMMary` creates a new Audio Summary view.

DISPlay:AUDio:MEASview:SELect

Selects an audio analysis view on the screen. The query returns the currently selected view.

Selecting a measurement optimizes it. Other measurements may be optimized as a side effect. Refer to the `DISPlay:WINDow:OPTimized:MEASurement?` query.

Conditions

Measurement views: Audio Spectrum, Audio Summary

Group

Display commands

Syntax

```
DISPlay:AUDio:MEASview:SELect { SPECtrum | SUMMary }
DISPlay:AUDio:MEASview:SELect?
```

Arguments

`SPECtrum` creates a new Audio Spectrum view.

`SUMMary` creates a new Audio Summary view.

If you attempt to open a view that is currently displayed on screen, the error (-200, "Execution error; Measurement is already running") will be returned.

Examples

`DISPLAY:AUDIO:MEASVIEW:SELECT SPECTrum` selects the Audio Spectrum

display.

DISPlay:AUDio:SPECtrum:FREQuency:AUTO (No Query Form)

Rescales the horizontal axis automatically to fit the waveform to the screen in the audio spectrum view.

Conditions

Measurement views: Audio Spectrum

Group

Display commands

Syntax

```
DISPlay:AUDio:SPECtrum:FREQuency:AUTO
```

Arguments

None

Examples

`DISPLAY:AUDIO:SPECTRUM:FREQUENCY:AUTO` rescales the horizontal scale automatically to fit the waveform to the screen.

DISPlay:AUDio:SPECtrum:FREQuency:[SCALe]:STARt

Sets or queries the start frequency (left edge) of the audio spectrum graph.

Conditions

Measurement views: Audio Spectrum

Group

Display commands

Syntax

```
DISPlay:AUDio:SPECtrum:FREQuency:[SCALe]:STARt <value>
DISPlay:AUDio:SPECtrum:FREQuency:[SCALe]:STARt?
```

Related Commands

DISPlay:AUDio:SPECtrum:FREQuency:[SCALe]:STOP (on page 678)

Arguments

`<value> ::= <NRf>` specifies the start frequency.

Examples

`DISPLAY:AUDIO:SPECTRUM:FREQUENCY:[SCALE]:START 0` sets the start frequency to 0 Hz.

DISPlay:AUDio:SPECtrum:FREQuency:[SCALe]:STOP

Sets or queries the stop frequency (right edge) of the audio spectrum graph.

Conditions

Measurement views: Audio Spectrum

Group

Display commands

Syntax

```
DISPlay:AUDio:SPECtrum:FREQuency:[SCALe]:STOP <value>
DISPlay:AUDio:SPECtrum:FREQuency:[SCALe]:STOP?
```

Related Commands

DISPlay:AUDio:SPECtrum:FREQuency:[SCALe]:STARt (on page 678)

Arguments

<value> ::= <NRf> specifies the stop frequency.

Examples

DISPLAY:AUDIO:SPECTRUM:FREQUENCY:[SCALE]:STOP 20E+3 sets the stop frequency to 20 kHz.

DISPlay:AUDio:SPECtrum:MARKer:SHOW:STATe

Determines whether to show or hide the readouts for the selected marker in the audio spectrum view.

Conditions

Measurement views: Audio Spectrum

Group

Display commands

Syntax

```
DISPlay:AUDio:SPECtrum:MARKer:SHOW:STATe { OFF | ON | 0 | 1 }
DISPlay:AUDio:SPECtrum:MARKer:SHOW:STATe
```

Arguments

OFF or 0 hides the readout for the selected marker in the view.

ON or 1 shows the readout for the selected marker in the view.

Examples

DISPLAY:AUDIO:SPECTRUM:MARKER:SHOW:STATE ON shows the readout for the selected marker in the audio spectrum view.

DISPlay:AUDio:SPECtrum:RESet:SCALe (No Query Form)

Resets the horizontal and vertical scales in the audio spectrum view.

Conditions

Measurement views: Audio Spectrum

Group

Display commands

Syntax

```
DISPlay:AUDio:SPECtrum:RESet:SCALe
```

Arguments

None

Examples

DISPLAY:AUDIO:SPECTRUM:RESET:SCALE resets the horizontal and vertical scales.

DISPlay:AUDio:SPECtrum:SCALe:LOG:STATe

Determines whether or not to set the horizontal axis logarithmic in the audio spectrum view.

Conditions

Measurement views: Audio Spectrum

Group

Display commands

Syntax

```
DISPlay:AUDio:SPECtrum:SCALe:LOG:STATe { OFF | ON | 0 | 1 }
DISPlay:AUDio:SPECtrum:SCALe:LOG:STATe?
```

Arguments

`OFF` or 0 sets the horizontal axis linear.

`ON` or 1 sets the horizontal axis logarithmic.

Examples

`DISPLAY:AUDIO:SPECTRUM:SCALE:LOG:STATE ON` sets the horizontal axis logarithmic in the audio spectrum view.

DISPlay:AUDio:SPECtrum:SHOW:NHARmonic:THReshold

Sets or queries the appearance of the non-harmonic threshold in the audio spectrum view.

Conditions

Measurement views: Audio Spectrum

Group

Display commands

Syntax

```
DISPlay:AUDio:SPECtrum:SHOW:NHARmonic:THReshold { SHADed | LINE | OFF }
DISPlay:AUDio:SPECtrum:SHOW:NHARmonic:THReshold?
```

Arguments

`SHADed` uses shading for the non-harmonic threshold area in the display.

`LINE` uses a line for the non-harmonic threshold area.

`OFF` turns off the non-harmonic threshold in the display.

Examples

`DISPLAY:AUDIO:SPECTRUM:SHOW:NHARMONIC:THRESHOLD SHADed` uses shading to identify the non-harmonic threshold area in the audio spectrum view.

DISPlay:AUDio:SPECtrum:TABLe:SHOW:STATe

Determines whether to show or hide the results table in the audio spectrum view.

Conditions

Measurement views: Audio Spectrum

Group

Display commands

Syntax

```
DISPlay:AUDio:SPECtrum:TABLe:SHOW:STATe { OFF | ON | 0 | 1 }
DISPlay:AUDio:SPECtrum:TABLe:SHOW:STATe?
```

Arguments

`OFF` or 0 disables (hides) the results table in the audio spectrum view.

`ON` or 1 shows the results table in the audio spectrum view.

Examples

`DISPLAY:AUDIO:SPECTRUM:TABLE:SHOW:STATE OFF` hides the results table in the audio spectrum view.

DISPlay:AUDio:SPECtrum:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions

Measurement views: Audio Spectrum

Group

Display commands

Syntax

```
DISPlay:AUDio:SPECtrum:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:AUDio:SPECtrum:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or 0 hides the graticule grid.

`ON` or 1 shows the graticule grid.

Examples

`DISPLAY:AUDIO:SPECTRUM:WINDOW:TRACE:GRATICULE:GRID:STATE OFF` hides the graticule grid on the screen.

DISPlay:AUDio:SPECtrum:Y:[SCALe]

Sets or queries the vertical scale in the audio spectrum view. The range of the scale is from 0.1 dB to 200 dB.

Conditions

Measurement views: Audio Spectrum

Group

Display commands

Syntax

```
DISPlay:AUDio:SPECtrum:Y:[SCALe] <value>
DISPlay:AUDio:SPECtrum:Y:[SCALe]?
```

Arguments

`<value> ::= <NRf>` specifies the vertical scale value in dB.

Examples

`DISPLAY:AUDIO:SPECTRUM:Y:[SCALE] 100` sets the vertical scale to 100 dB.

DISPlay:AUDio:SPECtrum:Y:[SCALe]:AUTO (No Query Form)

Rescales the vertical axis automatically in the audio spectrum view.

Conditions

Measurement views: Audio Spectrum

Group

Display commands

Syntax

```
DISPlay:AUDio:SPECtrum:Y:[SCALe]:AUTO
```

Arguments

None

Examples

`DISPLAY:AUDIO:SPECTRUM:Y:[SCALE]:AUTO` rescales the vertical axis automatically.

DISPlay:AUDio:SPECtrum:Y:[SCALe]:OFFSet

Sets or queries the vertical position (offset) in the audio spectrum view.

Conditions

Measurement views: Audio Spectrum

Group

Display commands

Syntax

```
DISPlay:AUDio:SPECtrum:Y:[SCALe]:OFFSet <value>
DISPlay:AUDio:SPECtrum:Y:[SCALe]:OFFSet?
```

Arguments

`<value> ::= <NRf>` specifies the vertical offset in dBm.

Examples

`DISPLAY:AUDIO:SPECTRUM:Y:[SCALE]:OFFSET 150` sets the vertical offset to 150 dBM.

DISPlay:AVTime[:MEASview<y>]:LEGend:STATe

Determines whether to show or hide the trace legend in the amplitude versus time view. The legend indicates the trace detection and function on the screen for each displayed trace.

Conditions

Measurement views: Amplitude versus Time

Group

Display commands

Syntax

```
DISPlay:AVTime[:MEASview<y>]:LEGend:STATe { OFF | ON | 0 | 1 }
DISPlay:AVTime[:MEASview<y>]:LEGend:STATe?
```

Arguments

OFF or 0 hides the trace legend.

ON or 1 shows the trace legend.

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

DISPLAY:AVTIME:MEASVIEW1:LEGEND:STATE ON shows the trace legend on the screen.

DISPlay:AVTime[:MEASview<y>]:MARKer:SHOW:STATe

Determines whether to show or hide the readout for the selected marker on the screen in the Amplitude versus Time measurement.

Conditions

Measurement views: Amplitude versus Time

Group

Display commands

Syntax

```
DISPlay:AVTime[:MEASview<y>]:MARKer:SHOW:STATe { OFF | ON | 0 | 1 }
DISPlay:AVTime[:MEASview<y>]:MARKer:SHOW:STATe?
```

Arguments

OFF or 0 hides the readout for the selected marker on screen.

ON or 1 shows the readout for the selected marker on screen.

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

DISPLAY:AVTIME:MEASVIEW1:MARKER:SHOW:STATE ON shows the readout for the selected marker on screen in the Amplitude versus Time Measurement View 1.

DISPlay:AVTime[:MEASview<y>]:SELect:FIRSt

Queries the trace whose measurement results are being displayed in the readout on the top and bottom of the view.

All traces are valid.

Conditions

Measurement views: Amplitude versus Time

Group

Display commands

Syntax

```
:DISPlay:AVTime[:MEASview<y>]:SELect:FIRSt
```

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

```
{ 0 | 1 }
```

0 indicates that the results are not being displayed in the readout for the first trace.

1 indicates that the results are being displayed in the readout for the first trace.

Examples

DISPLAY:AVTIME:MEASVIEW1:SELECT:FIRST? might return 1, indicating that the results are being displayed in the readout for Trace 1 in the Amplitude versus Time Measurement View 1.

DISPlay:AVTime[:MEASview<y>]:SELect:SECond

Sets or queries the trace whose measurement results are being displayed in the readout on the top and bottom of the view.

All traces are valid.

Conditions

Measurement views: Amplitude versus Time

Group

Display commands

Syntax

```
:DISPlay:AVTime[:MEASview<y>]:SELect:SECond
```

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

```
{ 0 | 1 }
```

0 indicates that the results are not being displayed in the readout for the first trace.

1 indicates that the results are being displayed in the readout for the first trace.

Examples

DISPLAY:AVTIME:MEASVIEW1:SELECT:FIRST? might return 1, indicating that the results are being displayed in the readout for Trace 2 in the Amplitude versus Time Measurement View 1.

DISPlay:AVTime[:MEASview<y>]:SELect:THIRd

Queries the trace whose measurement results are being displayed in the readout on the top and bottom of the view.

All traces are valid.

Conditions

Measurement views: Amplitude versus Time

Group

Trace commands

Syntax

```
:DISPlay:AVTime[:MEASview<y>]:SELect:THIRd
```

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

```
{ 0 | 1 }
```

0 indicates that the results are not being displayed in the readout for the first trace.

1 indicates that the results are being displayed in the readout for the first trace.

Examples

DISPLAY:AVTIME:MEASVIEW1:SELECT:FIRST? might return 1, indicating that the results are being displayed in the readout for Trace 3 in the Amplitude versus Time Measurement View 1.

DISPlay:AVTime[:MEASview<y>]:SELect:FOURth

Queries the trace whose measurement results are being displayed in the readout on the top and bottom of the view.

All traces are valid.

Conditions

Measurement views: Amplitude versus Time

Group

Trace commands

Syntax

```
:DISPlay:AVTime[:MEASview<y>]:SELect:FOURth
```

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

```
{ 0 | 1 }
```

0 indicates that the results are not being displayed in the readout for the first trace.

1 indicates that the results are being displayed in the readout for the first trace.

Examples

DISPLAY:AVTIME:MEASVIEW1:SELECT:FOUR? might return 1, indicating that the results are being displayed in the readout for Trace 4 in the Amplitude versus Time Measurement View 1.

DISPlay:AVTime[:MEASview<y>]:SELect:TIMestamp

Sets or queries the display of the time stamp in the ogram plot.

Conditions

Applicable measurement view must be active.

Group

Display commands

Syntax

`DISPlay:AVTime[:MEASview<y>]:SELect:TIMestamp {OFF | ON | 0 | 1}`

`DISPlay:AVTime[:MEASview<y>]:SELect:TIMestamp?`

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional. By default y is 1.

`OFF` or `0` removes or turns off the time stamp.

`ON` or `1` displays or turns on the time stamp.

Returns

`{OFF | ON | 0 | 1}`

Examples

`DISPlay:AVTime:MEASview1:SELect:TIMestamp?` queries the display status of the time stamp in the ogram plot view 1.

DISPlay:AVTime[:MEASview<y>]:READout[:STATe]

Determines whether to show or hide the measurement parameter readouts in the Amplitude versus Time view.

Conditions

Measurement views: Amplitude versus Time

Group

Display commands

Syntax

`DISPlay:AVTime[:MEASview<y>]:READout[:STATe] {OFF | ON | 0 | 1}`

`DISPlay:AVTime[:MEASview<y>]:READout[:STATe]?`

Arguments

`OFF` or `0` hides the parameter readouts.

`ON` or `1` shows the parameter readouts.

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1.

Examples

DISPLAY:AVTIME:MEASVIEW1:READOUT:STATE ON shows the parameter readouts on the screen.

DISPlay:AVTime[:MEASview<y>]:RESet (No Query Form)

Resets the horizontal and vertical scale to the default values described below in the Amplitude versus Time view.

Vertical offset = Reference level, Vertical scale = 100 dB,

Horizontal offset = Analysis offset, and Horizontal scale = Analysis length

Conditions

Measurement views: Amplitude versus Time

Group

Display commands

Syntax

DISPlay:AVTime[:MEASview<y>]:RESet

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

DISPLAY:AVTIME:MEASVIEW1:RESET resets the horizontal and vertical scale to the default values in the Amplitude versus Time Measurement View 1.

DISPlay:AVTime[:MEASview<y>]:TRIGger:LEVel:STATe

Determines whether to show or hide the power trigger level line on the screen in the Amplitude versus Time measurement.

Conditions

Measurement views: Amplitude versus Time

Group

Display commands

Syntax

DISPlay:AVTime[:MEASview<y>]:TRIGger:LEVel:STATe { OFF | ON | 0 | 1 }
DISPlay:AVTime[:MEASview<y>]:TRIGger:LEVel:STATe?

Arguments

OFF or 0 hides the power trigger level line.

ON or 1 shows the power trigger level line.

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

See Arguments.

Examples

DISPLAY:AVTIME:MEASVIEW1:TRIGGER:LEVEL:STATE ON shows the power trigger level line on the screen in the Amplitude versus Time Measurement View 1.

DISPlay:AVTime[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions

Measurement views: Amplitude versus Time

Group

Display commands

Syntax

```
DISPlay:AVTime[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON| 0 | 1 }
DISPlay:AVTime[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

OFF or 0 hides the graticule grid.

ON or 1 shows the graticule grid.

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

See Arguments.

Examples

DISPLAY:AVTIME:MEASVIEW1:WINDOW:TRACE:GRATICULE:GRID:STATE ON shows the graticule grid on the screen in the Amplitude versus Time Measurement View 1.

DISPlay:AVTime[:MEASview<y>]:X:RSCale (No Query Form)

Rescales the horizontal axis automatically to fit the waveform to the screen in the Amplitude versus Time display.

Conditions

Measurement views: Amplitude versus Time

Group

Display commands

Syntax

```
DISPlay:AVTime[:MEASview<y>]:X:RSCale
```

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

DISPLAY:AVTIME:MEASVIEW1:X:RSCALE rescales the horizontal scale automatically to fit the waveform to the screen in the Amplitude versus Time Measurement View 1.

DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:AUTO (No Query Form)

Sets the horizontal scale automatically to fit the waveform to the screen in the Amplitude versus Time view. Executing this command sets DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:AUTO:STATe ON.

Conditions

Measurement views: Amplitude versus Time

Group

Display commands

Syntax

DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:AUTO

Related Commands

DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:AUTO:STATe (on page 689)

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

DISPlay:AVTime:MEASview<y>:X:SCALe:AUTO sets the horizontal scale automatically to fit the waveform to the screen in the Amplitude versus Time Measurement View 1 graph.

DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:AUTO:STATe

Determines whether to set the horizontal scale automatically or manually.

Conditions

Measurement views: Amplitude versus Time

Group

Display commands

Syntax

DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:AUTO:STATe { OFF | ON | 0 | 1 }
DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:AUTO:STATe?

Arguments

OFF or 0 specifies that the horizontal scale is set manually. To set it, use the DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:FULL (on page 690) and DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:OFFSet (on page 691) commands.

ON or 1 specifies that the horizontal scale is set automatically.

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

See Arguments.

Examples

`DISPLAY:AVTIME:MEASVIEW1:X:SCALE:AUTO:STATE ON` specifies that the horizontal scale is set automatically in the Amplitude versus Time Measurement View 1 graph.

DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:FULL

Sets or queries the horizontal scale (full-scale time) of the Amplitude versus Time graph. Programming a specified scale sets `DISPlay:AVTime[:MEASview<y>]:X[:SCALe] AUTO:STATe OFF`.

Conditions

Measurement views: Amplitude versus Time

Group

Display commands

Syntax

```
DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:FULL <value>
DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:FULL?
```

Related Commands

DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:AUTO:STATe (on page 689)

DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:OFFSet (on page 691)

Arguments

`<value>::=<NRf>` specifies the horizontal scale in full-scale time. Use the DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:MAXimum? (on page 690) and DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:MINimum? (on page 691) queries to get the upper and lower limits of the setting range.

The parameter `<y>`=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

`<NRf>`

Examples

`DISPLAY:AVTIME:MEASVIEW1:X:SCALE:FULL 25.6us` sets the horizontal scale to 25.6 µs in the Amplitude versus Time Measurement View 1.

DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:MAXimum? (Query Only)

Queries the upper limit of the horizontal scale setting range.

Conditions

Measurement views: Amplitude versus Time

Group

Display commands

Syntax

```
DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:MAXimum?
```

Related Commands

DISPlay:AVTime[:MEASview<y>:]X[:SCALe]:FULL (on page 690)

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

<NRf> The upper limit of the horizontal scale setting range.

Examples

DISPLAY:AVTIME:MEASVIEW1:X:SCALE:MAXIMUM? might return 18.135E-3, indicating that the upper limit of the horizontal scale setting range is 18.135 ms in the Amplitude versus Time Measurement View 1.

DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:MINimum? (Query Only)

Queries the lower limit of the horizontal scale setting range.

Conditions

Measurement views: Amplitude versus Time

Group

Display commands

Syntax

```
DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:MINimum?
```

Related Commands

DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:FULL (on page 690)

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

<NRf> The lower limit of the horizontal scale setting range.

Examples

DISPLAY:AVTIME:MEASVIEW1:X:SCALE:MINIMUM? might return 10.0E-9, indicating that the lower limit of the horizontal scale setting range is 10.0 ns in the Amplitude versus Time Measurement View 1.

DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:OFFSet

Sets or queries the minimum horizontal value (left edge) of the Amplitude versus Time graph. Programming a specified offset sets DISPlay:AVTime[:MEASview<y>]:X[:SCALe] AUTO:STATe OFF.

Conditions

Measurement views: Amplitude versus Time

Group

Display commands

Syntax

```
DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:OFFSet <value>
DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:OFFSet?
```

Related Commands

DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:AUTO:STATe (on page 689)

DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:FULL (on page 690)

Arguments

`<value>::=<NRf>` specifies the minimum horizontal value. Use the DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MAXimum? (on page 692) and DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MINimum? (on page 693) queries to get the upper and lower limits of the setting range.

The parameter `<y>`=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

`<NRf>`

Examples

`DISPLAY:AVTIME:MEASVIEW1:X:SCALE:OFFSET 800ns` sets the minimum horizontal value to 800 ns in the Amplitude versus Time Measurement View 1 graph.

DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MAXimum? (Query Only)

Queries the upper limit of the horizontal offset setting range.

Conditions

Measurement views: Amplitude versus Time

Group

Display commands

Syntax

```
DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MAXimum?
```

Related Commands

DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:OFFSet (on page 691)

Arguments

The parameter `<y>`=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

<NRf> The upper limit of the horizontal offset setting range.

Examples

DISPLAY:AVTIME:MEASVIEW1:X:SCALE:OFFSET:MAXIMUM? might return -1.812E-3, indicating that the upper limit of the horizontal offset setting range is -1.812 ms in the Amplitude versus Time Measurement View 1.

DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MINimum? (Query Only)

Queries the lower limit of the horizontal offset setting range.

Conditions

Measurement views: Amplitude versus Time

Group

Display commands

Syntax

DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MINimum?

Related Commands

DISPlay:AVTime[:MEASview<y>]:X[:SCALe]:OFFSet (on page 691)

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

<NRf> The lower limit of the horizontal offset setting range.

Examples

DISPLAY:AVTIME:MEASVIEW1:X:SCALE:OFFSET:MINIMUM? might return -16.28E-3, indicating that the lower limit of the horizontal offset setting range is -16.28 ms in the Amplitude versus Time Measurement View 1.

DISPlay:AVTime[:MEASview<y>]:Y:RSCale (No Query Form)

Rescales the vertical axis automatically to fit the waveform to the screen in the Amplitude versus Time display.

Conditions

Measurement views: Amplitude versus Time

Group

Display commands

Syntax

DISPlay:AVTime[:MEASview<y>]:Y:RSCale

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

DISPLAY:AVTIME:MEASVIEW1:Y:RSCALE rescales the vertical scale automatically to fit the waveform to the screen in the Amplitude versus Time Measurement View 1 graph.

DISPlay:AVTime[:MEASview<y>]:Y[:SCALe]:FULL

Sets or queries the vertical range of the Amplitude versus Time graph.

Conditions

Measurement views: Amplitude versus Time

Group

Display commands

Syntax

```
DISPlay:AVTime[:MEASview<y>]:Y[:SCALe]:FULL <value>
DISPlay:AVTime[:MEASview<y>]:Y[:SCALe]:FULL?
```

Related Commands

DISPlay:AVTime[:MEASview<y>]:Y[:SCALe]:OFFSet (on page 694)

Arguments

<value>::=<NRf> specifies the vertical range. Range: 0.1 to 200 dB.

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

<NRf>

Examples

DISPLAY:AVTIME:MEASVIEW1:Y:SCALE:FULL 100 sets the vertical range to 100 dB in the Amplitude versus Time Measurement View 1 graph.

DISPlay:AVTime[:MEASview<y>]:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (the value at the top edge of the vertical axis) in the Amplitude versus Time graph.

Conditions

Measurement views: Amplitude versus Time

Group

Display commands

Syntax

```
DISPlay:AVTime[:MEASview<y>]:Y[:SCALe]:OFFSet <value>
DISPlay:AVTime[:MEASview<y>]:Y[:SCALe]:OFFSet?
```

Related Commands

DISPlay:AVTime[:MEASview<y>]:Y[:SCALe]:FULL (on page 694)

Arguments

<value>::=<NRf> specifies the vertical offset. Range: -170 to +50 dBm.

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

<NRf>

Examples

DISPLAY:AVTIME:MEASVIEW1:Y:SCALE:OFFSET -12.5 sets the vertical offset to -12.5 dBm in the Amplitude versus Time Measurement View 1 graph.

DISPlay:BIBEmissions:MARKer<x>:SHOW:STATe

Sets or queries to show or hide the readout for the selected marker Bluetooth InBand Emission view.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions

Measurement views: Bluetooth InBand Emission

Group

Display commands

Syntax

```
DISPlay:BIBEmissions:MARKer<x>:SHOW:STATe { OFF | ON | 0 | 1 }
DISPlay:BIBEmissions:MARKer<x>:SHOW:STATe?
```

Arguments

OFF or 0 hides the readout for the selected marker in the graph.

ON or 1 shows the readout for the selected marker in the graph.

Returns

See Arguments.

Examples

DISPLAY:BIBEMISSIONS:MARKER1:SHOW:STATE ON shows the readout for the Marker1 (M1) in the graph.

DISPlay:BIBEmissions:PLEVel:SHOW:STATe

Determines whether to show or hide the power levels in the Bluetooth InBand Emission view.

Conditions

Measurement views: Bluetooth InBand Emission

Group

Display commands

Syntax

```
DISPlay:BIBEmissions:PLEVel:SHOW:STATe { OFF | ON | 0 | 1 }
DISPlay:BIBEmissions:PLEVel:SHOW:STATe?
```

Arguments

`OFF` or 0 hides the power levels in the graph.

`ON` or 1 shows the power levels in the graph.

Returns

See Arguments.

Examples

`DISPLAY:BIBEMISSIONS:PLEVEL:SHOW:STATE ON` shows the power levels in the graph.

DISPlay:BIBEmissions:RESet:SCALe (No Query Form)

Resets the horizontal and vertical scale to the default values in the Bluetooth InBand Emission view.

Conditions

Measurement views: Bluetooth InBand Emission

Group

Display commands

Syntax

```
DISPlay:BIBEmissions:RESet:SCALe
```

Arguments

None

Examples

`DISPLAY:BIBEMISSIONS:RESET:SCALE` resets the horizontal and vertical scale to the default values.

DISPlay:BIBEmissions:WINDow:TRACe:GRATicule:GRID:STATe

Sets or queries the graticule hidden or showing state.

Conditions

Measurement views: Bluetooth InBand Emission

Group

Display commands

Syntax

```
DISPlay:BIBEmissions:WINDow:TRACe:GRATicule:GRID:STATe { OFF| ON | 0 | }
DISPlay:BIBEmissions:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or 0 hides the graticule grid.

`ON` or 1 shows the graticule grid.

Returns

See Arguments.

Examples

`DISPLAY:BIBEMISSIONS:WINDOW:TRACE:GRATICULE:GRID:STATE ON` shows the graticule grid on the screen.

DISPlay:BIBEmissions:X[:SCALe]

Sets or queries the horizontal range of the Bluetooth InBand Emission graph.

Conditions

Measurement views: Bluetooth InBand Emission

Group

Display commands

Syntax

```
DISPlay:BIBEmissions:X[:SCALe] <value>
DISPlay:BIBEmissions:X[:SCALe]?
```

Related Commands

DISPlay:BIBEmissions:Y[:SCALe]:OFFSet (on page 699)

Arguments

`<value>::=<NRf>` specifies the horizontal range.

Returns

`<NRf>`

Examples

`DISPLAY:BIBEMISSIONS:X:SCALE 10MHZ` sets the horizontal range to 10 MHz.

DISPlay:BIBEmissions:X[:SCALe]:AUTO (No Query Form)

Rescales the horizontal axis automatically to fit the waveform to the screen in the Bluetooth InBand Emission view.

Conditions

Measurement views: Bluetooth InBand Emission

Group

Display commands

Syntax

```
DISPlay:BIBEmissions:X[:SCALe]:AUTO
```

Arguments

None

Examples

`DISPLAY:BIBEMISSIONS:X:SCALE:AUTO` rescales the horizontal scale automatically to fit the waveform to the screen.

DISPlay:BIBEmissions:X[:SCALe]:OFFSet

Sets or queries the center frequency of the Bluetooth InBand Emission graph.

Conditions

Measurement views: Bluetooth InBand Emission

Group

Display commands

Syntax

```
DISPlay:BIBEmissions:X[:SCALe]:OFFSet <value>
DISPlay:BIBEmissions:X[:SCALe]:OFFSet?
```

Related Commands

DISPlay:BIBEmissions:X[:SCALe] (on page 697)

Arguments

`<value> ::= <NRf>` specifies the minimum horizontal value.

Returns

`<NRf>`

Examples

`DISPLAY:BIBEMISSIONS:X:SCALE:OFFSET 1.45GHZ` sets the center frequency to 1.45 GHz.

DISPlay:BIBEmissions:Y[:SCALe]

Sets or queries the vertical range of the Bluetooth InBand Emission graph.

Conditions

Measurement views: Bluetooth InBand Emission

Group

Display commands

Syntax

```
DISPlay:BIBEmissions:Y[:SCALe] <value>
DISPlay:BIBEmissions:Y[:SCALe]?
```

Related Commands

DISPlay:BIBEmissions:Y[:SCALe]:OFFSet (on page 699)

Arguments

`<value> ::= <NRf>` specifies the vertical range. Range: 0.1 to 200 dB.

Returns

`<NRf>`

Examples

`DISPLAY:BIBEMISSIONS:Y:SCALE 100` sets the vertical range to 100 dB.

DISPlay:BIBEmissions:Y[:SCALe]:AUTO (No Query Form)

Rescales the vertical axis automatically to fit the waveform to the screen in the Bluetooth InBand Emission view.

Conditions

Measurement views: Bluetooth InBand Emission

Group

Display commands

Syntax

```
DISPlay:BIBEmissions:Y[:SCALe]:AUTO
```

Arguments

None

Examples

`DISPLAY:BIBEMISSIONS:Y:SCALE:AUTO` rescales the vertical scale automatically to fit the waveform to the screen.

DISPlay:BIBEmissions:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (the value at the top edge of the vertical axis) in the Bluetooth InBand Emission graph.

Conditions

Measurement views: Bluetooth InBand Emission

Group

Display commands

Syntax

```
DISPlay:BIBEmissions:Y[:SCALe]:OFFSet <value>
DISPlay:BIBEmissions:Y[:SCALe]:OFFSet?
```

Related Commands

DISPlay:BIBEmissions:Y[:SCALe] (on page 698)

Arguments

`<value> ::= <NRf>` specifies the vertical offset. Range: -170 to +50 dBm.

Returns

`<NRf>`

Examples

`DISPLAY:BIBEMISSIONS:Y:SCALE:OFFSET -12.5` sets the vertical offset to -12.5 dBm.

DISPlay:BLUEtooth:CONSte:TRACe:GRATicule:GRID:STATe

Sets or queries the display state of Constellation graticule grid.

Conditions

Measurement views: Bluetooth Constellation

Group

Display commands

Syntax

```
DISPlay:BLUEtooth:CONSte:TRACe:GRATicule:GRID:STATe { 0 | 1 | OFF | ON }
DISPlay:BLUEtooth:CONSte:TRACe:GRATicule:GRID:STATe?
```

Arguments

`ON` or 1 turns on the grid.

`OFF` or 0 turns off the grid.

Returns

See Arguments.

Examples

`DISPlay:BLUEtooth:CONSte:TRACe:GRATicule:GRID:STATe 1` turns on the grid in the Constellation plot.

DISPlay:BLUEtooth:CONSte:WINDow:TRACe:GRATicule:GRID:STATe

Sets or queries whether to show or hide the graticule grid on the screen in the Bluetooth Constellation display.

Conditions

Measurement views: Bluetooth Constellation

Group

Display commands

Syntax

```
DISPlay:BLUEtooth:CONSte:WINDow:TRACe:GRATicule:GRID:STATe STATe { OFF | ON | 0 | 1 }
DISPlay:BLUEtooth:CONSte:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or 0 hides the graticule grid.

`ON` or 1 shows the graticule grid.

Returns

See Arguments.

Examples

`DISPlay:BLUEtooth:CONSte:WINDow:TRACe:GRATicule:GRID:STATe ON` shows the graticule grid on the screen.

DISPlay:BLUEtooth:EDIagram:WINDow:TRACe:GRATicule:GRID:STATe

Sets or queries graticule grid hidden or showing state in the Bluetooth eye diagram display.

Conditions

Measurement views: Bluetooth Eye diagram

Group

Display commands

Syntax

```
DISPlay:BLUEtooth:EDIagram:WINDow:TRACe:GRATicule:GRID:STATe { 0 | 1 | OFF | ON }
DISPlay:BLUEtooth:EDIagram:WINDow:TRACe:GRATicule:GRID: STATe?
```

Arguments

`OFF` or 0 hides the graticule grid.

`ON` or 1 shows the graticule grid.

Returns

See Arguments.

Examples

`DISPLAY:BLUETOOTH:EDIAGRAM:WINDOW:TRACE:GRATICULE:GRID:STATE ON` shows the graticule grid in the eye diagram.

DISPlay:BLUEtooth:EDIagram:Y[:SCALe]

Sets or queries the vertical range of the Bluetooth eye diagram.

Conditions

Measurement views: Bluetooth Eye diagram

Group

Display commands

Syntax

```
DISPlay:BLUEtooth:EDIagram:Y[:SCALe] <value>
DISPlay:BLUEtooth:EDIagram:Y[:SCALe]?
```

Arguments

`<value> ::= <NRf>` the vertical range (1 to 100)

Returns

`<NRf>`

Examples

`DISPlay:BLUEtooth:EDIagram:Y:SCALe 2.5` sets the vertical range to 2.5 in the eye diagram.

DISPlay:BLUEtooth:EDIagram:Y[:SCALe]:AUTO (No Query Form)

Sets the vertical scale automatically to fit the waveform in the Bluetooth eye diagram.

Conditions

Measurement views: Bluetooth Eye diagram

Group

Display commands

Syntax

```
DISPlay:BLUEtooth:EDIagram:Y[:SCALe]:AUTO
```

Arguments

None

Returns

`DISPLAY:BLUETOOTH:EDIAGRAM:Y:SCALE:AUTO` sets the vertical scale automatically to fit the waveform to the screen.

DISPlay:BLUEtooth:EDIagram:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (center point of the vertical axis) of the Bluetooth eye diagram.

Conditions

Measurement views: Bluetooth Eye diagram

Group

Display commands

Syntax

```
DISPlay:BLUEtooth:EDIagram:Y[:SCALe]:OFFSet <value>
DISPlay:BLUEtooth:EDIagram:Y[:SCALe]:OFFSet?
```

Arguments

`<value> ::= <NRf>` the minimum vertical scale offset, range (-50 to +50).

Returns

`<NRf>`

Examples

`DISPlay:BLUEtooth:EDIagram:Y:SCALe:OFFSet -0.5` sets the vertical offset to −0.5.

DISPlay:BLUEtooth:FDVTime:WINDow:TRACe:GRATicule:GRID:STATe

Shows or hides the graticule grid in the Bluetooth Frequency Deviation vs. Time display.

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Display commands

Syntax

```
DISPlay:BLUEtooth:FDVTime:WINDow:TRACe:GRATicule:GRID:STATe{OFF | ON | 0 | 1}
DISPlay:BLUEtooth:FDVTime:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or 0 hides the graticule grid.

`ON` or 1 shows the graticule grid.

Returns

See Arguments.

Examples

`DISPlay:BLUEtooth:FDVTime:WINDow:TRACe:GRATicule:GRID:STATe ON` specifies to show the graticule grid.

DISPlay:BLUEtooth:FDVTime:Y[:SCALe]:AUTO (No Query Form)

Rescales the vertical scale automatically to fit the waveform to the screen in the Bluetooth Frequency Deviation vs. Time display.

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Display commands

Syntax

```
DISPlay:BLUEtooth:FDVTime:Y[:SCALe]:AUTO
```

Arguments

None

Examples

`DISPlay:BLUEtooth:FDVTime:Y:SCALe:AUTO` rescales the vertical scale automatically to fit the waveform to the screen.

DISPlay:BLUEtooth:FDVTime:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (center point of the vertical axis) in the Bluetooth Frequency Deviation vs. Time display.

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Display commands

Syntax

```
DISPlay:BLUEtooth:FDVTime:Y[:SCALe]:OFFSet <value>
DISPlay:BLUEtooth:FDVTime:Y[:SCALe]:OFFSet?
```

Arguments

`<value> ::= <NRf>` the vertical offset. Range: -60 MHz to +60 MHz.

Returns

`<NRf>`

Examples

`DISPlay:BLUEtooth:FDVTime:Y:SCALe:OFFSet -14.5E+3` sets the vertical offset to -14.5 kHz.

DISPlay:BLUEtooth:MEASview:DELete (No Query Form)

Deletes the specified Bluetooth analysis view.

Conditions

Measurement views: All.

Group

Display commands

Syntax

```
DISPlay:BLUEtooth:MEASview:DELete
{CONS|SUMM|FDVT|EDI|BOBW|STAB|BIBE|DRIF}
```

Related Commands

DISPlay:BLUEtooth:MEASview:NEW (on page 704)

DISPlay:BLUEtooth:MEASview:SELect (on page 705)

Arguments

`CONS` deletes the BT Constellation view.

`SUMM` deletes the BT Summary view.

`FDVT` deletes the BT Freq Dev vs. Time view. EDI deletes the BT Eye Diagram view.

`BOBW` deletes the BT 20dB BW view. STAB deletes the BT Symbol Table view. BIBE deletes the BT InBand Emission view. DRIF deletes the BT Drift Table view.

If you attempt to delete a view that is not displayed on screen, the error (-200, "Execution error; Measurement not running" will be returned.

Examples

`DISPLAY:BLUETOOTH:MEASVIEW:DELETE CONS` deletes the BT Constellation view.

DISPlay:BLUEtooth:MEASview:NEW (No Query Form)

Displays a new measurement view for Bluetooth measurements.

Conditions

Measurement views: Bluetooth measurements.

Group

Display commands

Syntax

```
DISPlay:BLUEtooth:MEASview:NEW
{CONS|SUMM|FDVT|EDI|BOBW|STAB|BIBE|DRIF}
```

Arguments

`CONS` creates a new BT Constellation view.

`SUMM` creates a new BT Summary view.

`FDVT` creates a new BT Freq Dev vs. Time view.

`EDI` creates a new BT Eye Diagram view.

`BOBW` creates a new BT 20dB BW view.

`STAB` creates a new BT Symbol Table view.

`BIBE` creates a new BT InBand Emission view.

`DRIF` creates a new BT Drift Table view.

If you attempt to open a view that is currently displayed on screen, the error (-200, "Execution error; Measurement is already running") will be returned.

Examples

`DISPLAY:BLUETOOTH:MEASVIEW:NEW CONS` creates a new Bluetooth Constellation view.

DISPlay:BLUEtooth:MEASview:SELect

Selects a measurement view on the screen in the Bluetooth measurements. The query command returns the currently selected view.

Conditions

Measurement views: Bluetooth measurements.

Group

Display commands

Syntax

```
DISPlay:BLUEtooth:MEASview:SELect
{CONS|SUMM|FDVT|EDI|BOBW|STAB|BIBE|DRIF}
DISPlay:BLUEtooth:MEASview:SELect?
```

Arguments

`CONS` selects the BT Constellation view.

`SUMM` selects the BT Summary view.

`FDVT` selects the BT Freq Dev vs. Time view.

`EDI` selects the BT Eye Diagram view.

`BOBW` selects the BT 20dB BW view.

`STAB` selects the BT Symbol Table view.

`BIBE` selects the BT InBand Emission view.

`DRIF` selects the BT Drift Table view.

If you attempt to selects a view that is not displayed on screen, the error (-200, "Execution error; Measurement not running" will be returned.

Returns

See Arguments.

Examples

`DISPLAY:BLUETOOTH:MEASVIEW:SELECT EDI` selects the Bluetooth Eye Diagram view.

DISPlay:BOBW:SELected:BANDwidth

Sets or queries the bandwidth to measure in the Bluetooth 20 dB BW view.

Conditions

Measurement views: Bluetooth 20 dB BW

Group

Display commands

Syntax

```
DISPlay:BOBW:SELected:BANDwidth { XDBM | XDB }
DISPlay:BOBW:SELected:BANDwidth?
```

Arguments

`XDBM` selects the x dB bandwith to measrure.

`XDB` selects the x dB bandwith to measrure.

Returns

See Arguments.

Examples

`DISPlay:BOBW:SELected:BANDwidth XDB` selects the x dB bandwidth to measure.

DISPlay:CCDF:LEGend:STATe

Determines whether to show or hide the trace legend in the CCDF view. The legend indicates the trace detection and function on the screen for each displayed CCDF trace.

Conditions

Measurement views: CCDF

Group

Display commands

Syntax

```
DISPlay:CCDF:LEGend:STATe { OFF | ON | 0 | 1 }
DISPlay:CCDF:LEGend:STATe?
```

Arguments

`OFF` or 0 hides the trace legend.

`ON` or 1 shows the trace legend.

Returns

See Arguments.

Examples

`DISPlay:CCDF:LEGend:STATe ON` shows the trace legend on the screen.

DISPlay:CCDF:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions

Measurement views: CCDF

Group

Display commands

Syntax

```
DISPlay:CCDF:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:CCDF:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or 0 hides the graticule grid.

`ON` or 1 shows the graticule grid.

Returns

See Arguments.

Examples

`DISPlay:CCDF:WINDow:TRACe:GRATicule:GRID:STATe ON` shows the graticule grid on the screen in the CCDF view.

DISPlay:CONSte[:MEASview<y>]:MPHase

Sets or queries the multiplication constant of the phase multiplication constellation display for a CPM signal. This command is valid when `[SENSe]:DDEMod:MODulation:TYPE` is set to CPM.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Constellation

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Constellation).

Group

Display commands

Syntax

```
DISPlay:CONSte[:MEASview<y>]:MPHase { P1 | P2 | P4 | P8 | P16 | P32 }
DISPlay:CONSte[:MEASview<y>]:MPHase?
```

Arguments

The following table shows the arguments and phase multiplier.

Phase multiplication

Argument	Phase multiplier
P1	1
P2	2
P4	4
P8	8
P16	16
P32	32

Returns

See Arguments.

Examples

`DISPlay:CONSte[:MEASview<y>]:MPHase P8` sets the phase multiplier to 8 for measview 2 of constellation.

DISPlay:CONSte[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Constellation

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Constellation).

Group

Display commands

Syntax

```
DISPlay:CONSte[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:CONSte[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or 0 hides the graticule grid.

`ON` or 1 shows the graticule grid.

Returns

See Arguments.

Examples

`DISPlay:CONSte:MEASview2:WINDow:TRACe:GRATicule:GRID:STATE ON` shows the graticule grid on the screen for measview 2 of constellation.

DISPlay:DDEMod[:MEASview<y>]:DELete (No query form)

Deletes the measurement view in the general purpose digital modulation measurements.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: General purpose digital modulation

Group

Display commands

Syntax

```
DISPlay:DDEMod[:MEASview<y>]:DELete { CONSte | DIQVtime | EDIagram | EVM | FDVTime | MERRor |
    PERRor | SIGNalqual | STABle | TDIagram }
```

Arguments

The following table lists the arguments. The arguments are the string type.

Modulation measurement views

Argument	View
CONSte	Constellation
DIQVtime	Demodulated I&Q versus Time
EDIagram	Eye diagram
EVM	EVM (Error Vector Magnitude) versus Time
FDVTime	Frequency deviation versus Time
MERRor	Magnitude error versus Time
PERRor	Phase error versus Time
SIGNalqual	Signal quality
STABle	Symbol table
TDIagram	Trellis diagram

If you attempt to delete a view that is not displayed on screen, the error (-200, "Execution error; Measurement not running") will be returned.

Examples

`DISPlay:DDEMod:MEASview1:DELete CONSte` deletes the constellation view with measurement view 1.

DISPlay:DDEMod:MEASview<y>[:SOURce<x>]:NEW (No query form)

Displays a new measurement view in the general purpose digital modulation measurements.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: General purpose digital modulation

Group

Display commands

Syntax

```
DISPlay:DDEMod:MEASview<y>[:SOURce<x>]:NEW { CONSte | DIQVtime | EDIagram | EVM | FDVTime | MERRor
    | PERRor | SIGNalqual | STABle | TDIagram }
```

Arguments

<x> = 1 to 8, representing Range of Source.

If you attempt to open a view that is currently displayed on screen, the error (-200, "Execution error; Measurement is already running") will be returned.

Examples

`DISPLay:DDEMod:MEASview:SOURce1:NEW CONSte` creates the constellation view with measurement view 1 for source1.

DISPlay:DDEMod[:MEASview<y>]:SELect

Selects a measurement view in the general purpose digital modulation measurements on the screen. The query command returns the currently selected view.

Selecting a measurement optimizes it. Other measurements may be optimized as a side effect. Refer to `DISPlay:WINDow:OPTimized:MEASurement?` query.

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Conditions

Measurement views: General purpose digital modulation

Group

Display commands

Syntax

```
DISPlay:DDEMod[:MEASview<y>]:SELect { CONSte | DIQVtime | EDIagram | EVM | FDVTime | MERRor |
    PERRor | SIGNalqual | STABle | TDIagram }
DISPlay:DDEMod[:MEASview<y>]:SELect?
```

Arguments

If you attempt to select a view that is not displayed on screen, the error (-200, "Execution error; Measurement not running") will be returned.

Examples

`DISPlay:DDEModMEASview1:SELect CONSte` selects the constellation view with measurement view 1.

DISPlay:DDEMod[:SOURce<x>]:RADix

Sets or queries the base of symbols of the measurement source. This command is effective in the symbol table.

Conditions

Measurement views: Symbol table

Source keyword is optional, if not present Source 1 will be default source.

Group

Display commands

Syntax

```
DISPlay:DDEMod[:SOURce<x>]:RADix { BINary | HEXadecimal }
DISPlay:DDEMod[:SOURce<x>]:RADix?
```

Arguments

<x> = 1 to 8, representing Range of Source.

`BINary` selects binary notation.

`HEXadecimal` selects hexadecimal notation.

Examples

`DISPLay:DDEMod:SOURce1:RADix BINary` selects the binary notation of the measurement source symbol table.

DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]

Sets or queries the measurement source horizontal scale (full-scale time) for the time measurements in the general purpose digital modulation analysis. Programming a specified scale sets `DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:AUTO:STATe OFF`.

Conditions

Measurement views: EVM versus Time, Magnitude error versus Time, Phase error versus Time

Source keyword is optional, if not present Source1 will be default source.

Group

Display commands

Syntax

```
DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]<value>
DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]?
```

Related Commands

DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:AUTO:STATe (on page 712)

DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:OFFSet (on page 714)

Arguments

<x> = 1 to 8, representing Range of Source.

`<value>::=<NRf>` specifies the horizontal scale in full-scale time.

Use the DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:MAXimum and DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:MINimum? queries to get the upper and lower limits of the setting range. The unit can be changed by the [SENSe]:DDEMod[:SOURce<x>]:TIMe:UNITs command.

Returns

<NRf>

Examples

DISPlay:DDEMod:SOURce1:X:SCALe 1.5 µs, sets the horizontal scale of the measurement source to 1.5 µs.

DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:AUTO (No query form)

Sets the horizontal scale of the measurement source automatically to fit the waveform to the screen in the general purpose digital modulation analysis. Executing this command sets DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:AUTO:STATeON.

Conditions

Measurement views: EVM versus Time, Magnitude error versus Time, Phase error versus Time

Source keyword is optional, if not present Source1 will be default source.

Group

Display commands

Syntax

DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:AUTO

Related Commands

DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:AUTO:STATe (on page 712)

Arguments

<x> = 1 to 8, representing Range of Source.

Examples

DISPlay:DDEMod:SOURCE1:X:SCALe:AUTO sets the horizontal scale of the measurement source1 automatically to fit the waveform to the screen.

DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:AUTO:STATe

Determines whether to set the horizontal scale of the measurement source automatically or manually.

Conditions

Measurement views: EVM versus Time, Magnitude error versus Time, Phase error versus Time

Source keyword is optional, if not present Source1 will be default source.

Group

Display commands

Syntax

DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:AUTO:STATe { OFF | ON | 0 | 1 }
DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:AUTO:STATe?

Arguments

<x> = 1 to 8, representing Range of Source.

`OFF` or 0 specifies that the horizontal scale is set manually. To set it, use the `DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]` and `DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:OFFSet` commands.

`ON` or 1 specifies that the horizontal scale is set automatically.

Returns

ON or 1.

OFF or 0.

Examples

`DISPlay:DDEMod:SOURce1:X:SCALe:AUTO:STATeON` specifies that the horizontal scale of the measurement source1 is set automatically.

DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:MAXimum? (Query only)

Queries the upper limit of the measurement source horizontal scale setting range.

Conditions

Measurement views: EVM versus Time, Magnitude error versus Time, Phase error versus Time

Source keyword is optional, if not present Source1 will be default source.

Group

Display commands

Syntax

`DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:MAXimum?`

Returns

`<NRf>` The upper limit of the horizontal scale setting range.

The unit can be changed by the `[SENSe]:DDEMod[:SOURce<x>]:TIMe:UNITs` command.

Examples

`DISPlay:DDEMod:SOURce1:X:SCALe:MAXimum?` might return 18.135E-3, indicating that the upper limit of the measurement source1 horizontal scale setting range is 18.135 ms.

DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:MINimum? (Query only)

Queries the lower limit of the measurement source horizontal scale setting range.

Conditions

Measurement views: EVM versus Time, Magnitude error versus Time, Phase error versus Time

Source keyword is optional, if not present Source1 will be default source.

Group

Display commands

Syntax

```
DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:MINimum?
```

Returns

<NRf> The lower limit of the horizontal scale setting range.

The unit can be changed by the [SENSe]:DDEMod[:SOURce<x>]:TIMe:UNITs command.

Examples

DISPlay:DDEMod:SOURce1:X:SCALe:MINimum? might return 10.0E-9, indicating that the lower limit of the measurement source1 horizontal scale setting range is 10.0 ns.

DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]: OFFSet

Sets or queries the minimum horizontal value of the measurement source (left edge) for the time measurements in the general purpose digital modulation analysis.

Programming a specified offset sets DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:AUTO:STATe OFF.

Conditions

Measurement views: EVM versus Time, Magnitude error versus Time, Phase error versus Time

Group

Display commands

Syntax

```
DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:OFFSet<value>
DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:OFFSet?
```

Related Commands

DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:AUTO:STATe (on page 712)

DISPlay:DDEMod[:SOURce<x>]:X[:SCALe] (on page 711)

Arguments

<x> = 1 to 8, representing Range of Source.

<value>::=<NRf> specifies the minimum horizontal value.

Use the DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:OFFSet:MAXimum? and DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:OFFSet:MINimum? queries to get the upper and lower limits of the setting range.

The unit can be changed by the [SENSe]:DDEMod[:SOURce<x>]:TIMe:UNITs command.

Returns

<NRf>

Examples

DISPlay:DDEMod:SOURce1:X:SCALe:OFFSet 20.075 µs, sets the minimum horizontal value of the measurement source to 20.075 µs.

DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:OFFSet:MAXimum? (Query only)

Queries the measurement source upper limit of the horizontal offset setting range.

Conditions

Measurement views: EVM versus Time, Magnitude error versus Time, Phase error versus Time

Source keyword is optional, if not present Source1 will be default source.

Group

Display commands

Syntax

`DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:OFFSet:MAXimum?`

Returns

`<NRf>` The upper limit of the horizontal offset setting range. The unit can be changed by the `[SENSe]:DDEMod[:SOURce<x>]:TIMe:UNITs` command.

Examples

`DISPLay:DDEMod:SOURce1:X:SCALe:OFFSet:MAXimum?` might return -1.812E-3, indicating that the measurement source 1 upper limit of the horizontal offset setting range is -1.812 ms.

DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:OFFSet:MINimum? (Query only)

Queries the measurement source lower limit of the horizontal offset setting range.

Conditions

Measurement views: EVM versus Time, Magnitude error versus Time, Phase error versus Time

Source keyword is optional, if not present Source1 will be default source.

Group

Display commands

Syntax

`DISPlay:DDEMod:X[:SCALe]:OFFSet:MINimum?`

Returns

`<NRf>` The lower limit of the horizontal offset setting range. The unit can be changed by the `[SENSe]:DDEMod[:SOURce<x>]:TIMe:UNITs` command.

Examples

`DISPLay:DDEMod:SOURCE1:X:SCALe:OFFSet:MINImum?` might return -16.28E-3, indicating that the measurement source1 lower limit of the horizontal offset setting range is -16.28 ms.

DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:RESet (No query form)

Presets the horizontal scale of the measurement source to the default value for the time measurements in the general purpose digital modulation analysis.

Conditions

Measurement views: EVM versus Time, Magnitude error versus Time, Phase error versus Time

Source keyword is optional, if not present Source1 will be default source.

Group

Display commands

Syntax

```
DISPlay:DDEMod[:SOURce<x>]:X[:SCALe]:RESet
```

Arguments

<x> = 1 to 8, representing Range of Source.

Examples

`DISPLay:DDEMod:SOURCE1:X:SCALe:RESet` presets the horizontal scale of the measurement source1 to the default value.

DISPlay:DIAGram:X[:SCALe]

Sets or queries the horizontal range for the eye and trellis diagrams in the general purpose digital modulation analysis.

Conditions

Measurement views: Eye diagram, Trellis diagram

Group

Display commands

Syntax

```
DISPlay:DIAGram:X[:SCALe] <value>
DISPlay:DIAGram:X[:SCALe]?
```

Arguments

`<value>::=<NR1>` specifies the horizontal range.

Range: 1 to 16 symbols.

Returns

`<NR1>`

Examples

`DISPLAY:DIAGRAM:X:SCALE 3` sets the horizontal range to 3 symbols in the eye and trellis diagrams.

DISPlay:DIAGram:X[:SCALe]:RESet (No Query Form)

Presets the horizontal scale to the default value for the eye and trellis diagrams in the general purpose digital modulation analysis.

Conditions

Measurement views: Eye diagram, Trellis diagram

Group

Display commands

Syntax

```
DISPlay:DIAGram:X[:SCALe]:RESet
```

Arguments

None

Examples

`DISPLAY:DIAGRAM:X:SCALE:RESET` presets the horizontal scale to the default value for the eye and trellis diagrams.

DISPlay:DIQVtime[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Demodulated I&Q versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod I&Q versus Time).

Group

Display commands

Syntax

```
DISPlay:DIQVtime[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:DIQVtime[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or 0 hides the graticule grid.

`ON` or 1 shows the graticule grid.

Returns

See Arguments.

Examples

`DISPlay:DIQVtime:MEASview1:WINDow:TRACe:GRATicule:GRID:STATe ON` shows the graticule grid on the screen in the Demodulated I&Q versus Time view 1.

DISPlay:DIQVtime[:MEASview<y>]:Y[:SCALe]

Sets or queries the vertical range of the Demodulated I&Q versus Time graph.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Demodulated I&Q versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod I&Q versus Time).

Group

Display commands

Syntax

```
DISPlay:DIQVtime[:MEASview<y>]:Y[:SCALe] <value>
DISPlay:DIQVtime[:MEASview<y>]:Y[:SCALe]?
```

Arguments

`<value>::=<NRf>` specifies the vertical range. Range: 1 µV to 10 V.

Returns

`<NRf>`

Examples

`DISPlay:DIQVtime1:MEASview1:Y:SCALe 2.5` sets the vertical range to 2.5 V in the Demodulated I&Q versus Time measurement view 1.

DISPlay:DIQVtime[:MEASview<y>]:Y[:SCALe]:AUTO (No query form)

Sets the vertical scale automatically to fit the waveform to the screen in the Demodulated I & Q versus Time display.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Demodulated I&Q versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod I&Q versus Time).

Group

Display commands

Syntax

```
DISPlay:DIQVtime[:MEASview<y>]:Y[:SCALe]:AUTO
```

Arguments

None

Examples

`DISPlay:DIQVtime:MEASview1:Y:SCALe:AUTO` sets the vertical scale automatically to fit the waveform to the screen measurement view 1.

DISPlay:DIQVtime[:MEASview<y>]:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (center point of the vertical axis) of the Demodulated I&Q versus Time graph.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Demodulated I&Q versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod I&Q versus Time).

Group

Display commands

Syntax

```
DISPlay:DIQVtime[:MEASview<y>]:Y[:SCALe]:OFFSet <value>
DISPlay:DIQVtime[:MEASview<y>]:Y[:SCALe]:OFFSet?
```

Arguments

`<value>::=<NRf>` specifies the vertical offset. Range: -5 to +5 V.

Returns

`<NRf>`

Examples

`DISPlay:DIQVtime:MEASview1:Y:SCALe:OFFSet -0.5` sets the vertical offset to -0.5 V in the Demodulated I&Q versus Time measurement view 1.

DISPlay:DPX:DGRam:TIME[:SCALe]:OFFSet:DIVision

Sets or queries the DPXogram vertical time offset in divisions.

Conditions

Measurement views: DPX spectrum

Group

Display commands

Syntax

```
DISPlay:DPX:DGRam:TIME[:SCALe]:OFFSet:DIVision <value>
DISPlay:DPX:DGRam:TIME[:SCALe]:OFFSet:DIVision?
```

Arguments

`<value> ::= <NRf>` vertical offset value in divisions.

Returns

<NRf>

Examples

`DISPLAY:DPX:DGRAM:TIME:SCALE:OFFSET:DIVISION 10` sets the vertical time off set to 10 divisions.

DISPlay:DPX:DGRam:TIME[:SCALe]:PDIVision

Sets or queries the DPXogram vertical time scale per division.

Conditions

MMeasurement views: DPX spectrum

Group

Display commands

Syntax

```
DISPlay:DPX:DGRam:TIME[:SCALe]:PDIVision <value>
DISPlay:DPX:DGRam:TIME[:SCALe]:PDIVision?
```

Arguments

`<value>::<NRf>` vertical value in divisions.

Returns

`<NRf>`

Examples

`DISPLAY:DPX:DGRAM:TIME:SCALE:PDIVISION 10` sets the vertical time scale to 10 seconds per division.

DISPlay:DPX:DGRam:TIME[:SCALe]:RESet (No Query Form)

Presets the time scale and offset to the default value for the DPXogram measurement.

Conditions

Measurement views: DPX spectrum

Group

Display commands

Syntax

```
DISPlay:DPX:DGRam:TIME[:SCALe]:RESet
```

Arguments

None

Examples

`DISPLAY:DPX:DGRAM:TIME:SCALE:RESET` presets the time scale and offset to the default value.

DISPlay:DPX:DGRam:TSTamp:STATe

Determines whether to display the time stamp readout in the DPXogram display.

Conditions

Measurement views: DPX spectrum

Group

Display commands

Syntax

```
DISPlay:DPX:DGRam:TSTamp:STATe { OFF | ON | 0 | 1 }
DISPlay:DPX:DGRam:TSTamp:STATe?
```

Arguments

`OFF` or 0 hides the time stamp readout.

`ON` or 1 shows the time stamp readout.

Returns

See Arguments.

Examples

`DISPLAY:DPX:DGRAM:TSTAMP:STATE 0` hides the time stamp readout in the DPXogram display.

DISPlay:DPX:DGRam:Y[:SCALe]:AUTO (No Query Form)

Rescales the height axis automatically to fit the waveform to the screen in the DPXogram display.

Conditions

Measurement views: DPX spectrum

Group

Display commands

Syntax

`DISPlay:DPX:DGRam:Y[:SCALe]:AUTO`

Arguments

None

Examples

`DISPLAY:DPX:DGRAM:Y:AUTO` rescales the height scale automatically to fit the waveform to the DPXogram display.

DISPlay:DPX:DGRam:Y[:SCALe]:RESet (No Query Form)

Resets the height scale of the DPXogram display to the default values: Height position = 0 dB and Height scale = 100 dB.

Conditions

Measurement views: DPX spectrum

Group

Display commands

Syntax

`DISPlay:DPX:DGRam:Y[:SCALe]:RESet`

Arguments

None

Examples

`DISPLAY:DPX:DGRAM:Y:RESET` resets the height scale of the DPXogram display.

DISPlay:DPX:LEGend:STATe

Determines whether to show or hide the trace legend on the display. The legend indicates the trace detection and function on the screen for each displayed trace.

Conditions

Measurement views: DPX spectrum

Group

Display commands

Syntax

```
DISPlay:DPX:LEGend:STATe { OFF | ON | 0 | 1 }
DISPlay:DPX:LEGend:STATe?
```

Arguments

`OFF` or 0 hides the trace legend.

`ON` or 1 shows the trace legend.

Returns

See Arguments.

Examples

`DISPLAY:DPX:LEGEND:STATE ON` shows the trace legend on the screen.

DISPlay:DPX:PHASe:Y[:SCALe]:AXIS

Sets or queries the vertical axis representation.

Conditions

Measurement views: DPX Phase

Group

Display commands

Syntax

```
DISPlay:DPX:PHASe:Y[:SCALe]:AXIS { MODulopi | CONTinuous }
DISPlay:DPX:PHASe:Y[:SCALe]:AXIS?
```

Arguments

`MODulopi` (modulo π) shows the phase constrained within ±180° along the vertical axis.

`CONTinuous` shows the phase as continuous quantity along the vertical axis.

Returns

See Arguments.

Examples

`DISPLAY:DPX:PHASE:Y[:SCALE]:AXIS MODulopi` selects modulo π representation for the vertical axis.

DISPlay:DPX:PHASe:Y[:SCALe]:AXIS:OFFSet

Sets or queries the vertical offset in the DPX Phase view.

Conditions

Measurement views: DPX Phase

Group

Display commands

Syntax

```
DISPlay:DPX:PHASe:Y[:SCALe]:AXIS:OFFSet <value>
DISPlay:DPX:PHASe:Y[:SCALe]:AXIS:OFFSet?
```

Arguments

`<value>::<NRf>` specifies the vertical offset. Range: −180° to +180°.

Returns

`<NRf>`

Examples

`DISPLAY:DPX:PHASE:Y[:SCALE]:AXIS:OFFSET -158.5` sets the vertical offset to −158.5° in the view.

DISPlay:DPX:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions

Measurement views: DPX spectrum

Group

Display commands

Syntax

```
DISPlay:DPX:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:DPX:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or 0 hides the graticule grid.

`ON` or 1 shows the graticule grid.

Returns

See Arguments.

Examples

`DISPLAY:DPX:WINDOW:TRACE:GRATICULE:GRID:STATE ON` shows the graticule grid on the screen in the DPX spectrum view.

DISPlay:DPX:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (the value at the top edge of the vertical axis) of the DPX spectrum view.

Conditions

Measurement views: DPX spectrum

Group

Display commands

Syntax

```
DISPlay:DPX:Y[:SCALe]:OFFSet <value> DISPlay:DPX:Y[:SCALe]:OFFSet?
```

Arguments

`<value>::<NRf>` specifies the vertical offset. Range: -270 to +150 dBm.

The amplitude unit is set by the [SENSe]:POWer:UNITs command (default: dBm).

You can omit the unit in the argument. When you include the unit in the argument, only dBm is allowed. For the other amplitude units, omit the unit in the argument.

Returns

`<NRf>`

Examples

`DISPLAY:DPX:Y[:SCALE]:OFFSET -12.5dBm` sets the vertical offset to -12.5 dBm.

DISPlay:DPX:Y[:SCALe]:PDIVision

Sets or queries the vertical scale (per division) in the DPX spectrum view.

Conditions

Measurement views: DPX spectrum

Group

Display commands

Syntax

```
DISPlay:DPX:Y[:SCALe]:PDIVision <value>
DISPlay:DPX:Y[:SCALe]:PDIVision?
```

Related Commands

[SENSe]:POWer:UNITs (on page 2269)

Arguments

`<value>::<NRf>` specifies the vertical scale (per division).

Range: 2 to 20 dB/div.

Returns

`<NRf>`

Examples

`DISP:DPX:Y:SCAL:PDIV 0.5` sets the vertical scale to 0.5 dB/div.

DISPlay:EDIagram[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen in the eye diagram.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Eye diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Eye diagram).

Group

Display commands

Syntax

```
DISPlay:EDIagram[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:EDIagram[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or 0 hides the graticule grid.

`ON` or 1 shows the graticule grid.

Returns

See Arguments.

Examples

`DISPlay:EDIagram:MEASview2:WINDow:TRACe:GRATicule:GRID:STATe ON` shows the graticule grid on the screen in the eye diagram for measview2.

DISPlay:EDIagram[:MEASview<y>]:Y[:SCALe]

Sets or queries the vertical range of the Eye diagram.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Eye diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Eye diagram).

Group

Display commands

Syntax

```
DISPlay:EDIagram[:MEASview<y>]:Y[:SCALe]<value>
```

DISPlay:EDIagram[:MEASview<y>]:Y[:SCALe]?

Arguments

<value>::<NRf> specifies the vertical range. Range: 1 µ to 100 (unitless).

Returns

<NRf>

Examples

DISPlay:EDIagram:MEASview2:Y:SCALe 2.5 sets the vertical range to 2.5 in the eye diagram for measview2.

DISPlay:EDIagram[:MEASview<y>]:Y[:SCALe]:AUTO (No query form)

Sets the vertical scale automatically to fit the waveform to the screen in the eye diagram.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Eye diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Eye diagram).

Group

Display commands

Syntax

DISPlay:EDIagram[:MEASview<y>]:Y[:SCALe]:AUTO

Arguments

None

Examples

DISPLay:EDIagram:MEASview2:Y:SCALe:AUTO sets the vertical scale automatically to fit the waveform to the screen for measview2 of the Eye diagram.

DISPlay:EDIagram[:MEASview<y>]:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (center point of the vertical axis) of the eye diagram.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Eye diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Eye diagram).

Group

Display commands

Syntax

```
DISPlay:EDIagram[:MEASview<y>]:Y[:SCALe]:OFFSet<value>
DISPlay:EDIagram[:MEASview<y>]Y[:SCALe]:OFFSet?
```

Arguments

`<value>::<NRf>` specifies the minimum vertical value.

Range: -50 to +50 (unitless).

Returns

<NRf>

Examples

`DISPLay:EDIagram:MEASview2:Y:SCALe:OFFSet -0.5` sets the vertical offset to -0.5 in the eye diagram measview2.

DISPlay:EMC:INCLude:MARGin

Sets or queries to display or hide the Margin by enabling or disabling the check box in the Prefs tab of the EMC Settings control panel.

Conditions

Measurement view: EMC - EMI Display

Group

Display commands

Syntax

```
DISPlay:EMC:INCLude:MARGin { 0 | OFF | 1 | ON}
DISPlay:EMC:INCLude:MARGin?
```

Arguments

`0` or OFF hides the Margin

`1` or ON displays the margin

Returns

<Boolean> 0, 1

Examples

`DISPLAY:EMC:INCLUDE:MARGIN` returns 1, indicating that the Margin is displayed.

DISPlay:EMC:MARKer:SHOW:STATe

Sets or queries to show or hide the marker readout in the EMC view.

Conditions

Measurement view: EMC - EMI Display

Group

Display commands

Syntax

```
DISPlay:EMC:MARKer:SHOW:STATe { 0 | 1 | ON | OFF }
DISPlay:EMC:MARKer:SHOW:STATe?
```

Arguments

`OFF` or `0` hides the marker readout

`1` or `ON` shows the marker readout

Returns

<Boolean> 0, 1

Examples

`DISPLAY:EMC:MARKER:SHOW:STATE ON` shows the readout for the marker on the display.

DISPlay:EMC:RESet:SCALe (No Query Form)

Resets the horizontal and vertical scale to the default values in the EMC view.

Conditions

Measurement view: EMC - EMI Display

Group

Display commands

Syntax

```
DISPlay:EMC:RESet:SCALe
```

Arguments

None

Returns

None

Examples

`DISPLAY:EMC:RESET:SCALE` resets the horizontal and vertical scale to the default values.

DISPlay:EMC:SCALe:LOG:STATe

Determines whether or not to set the horizontal axis logarithmic in the EMC view.

Conditions

Measurement view: EMC - EMI Display

Group

Display commands

Syntax

```
DISPlay:EMC:SCALe:LOG:STATe { 0 | OFF | 1 | ON}
DISPlay:EMC:SCALe:LOG:STATe?
```

Arguments

OFF or 0 sets the horizontal axis linear (default).

ON or 1 sets the horizontal axis logarithmic.

Returns

<Boolean> 0, 1

Examples

DISPLAY:EMC:SCALE:LOG:STATE ON sets the horizontal axis logarithmic in the EMC view.

DISPlay:EMC:SELect:SPOT:NUMBer

Sets or queries the status of current spot number in the EMC view.

Conditions

Measurement view: EMC - EMI Display

Group

Display commands

Syntax

```
DISPlay:EMC:SELect:SPOT:NUMBer <value>
DISPlay:EMC:SELect:SPOT:NUMBer?
```

Arguments

<value> = <NR1>

Returns

<NR1>

Examples

DISPLAY:EMC:SELECT:SPOT:NUMBER might return +2, indicating that spot 2 is made current spot and it is enabled. (+ represents enabling and – represents disabling.)

DISPlay:EMC:SHOW:LIMit

Sets or queries the displayed limit format in EMC View.

Conditions

Measurement view: EMC - EMI Display

Group

Display commands

Syntax

```
DISPlay:EMC:SHOW:LIMit {LINE|SHAD|OFF}
DISPlay:EMC:SHOW:LIMit?
```

Arguments

LINE – Sets the limit in Line format

SHAD – Sets the limit in Shaded format

OFF – Hides the Limit

Returns

<String> Either LINE or SHAD or OFF

Examples

DISPLAY:EMC:SHOW:LIMIT might return LINE, indicating that the limit is displayed in line format.

DISPlay:EMC:SHOW:LIMit:LEGend

Sets or queries to display or hide the limit legends and label on the plot by enabling or disabling the check box in the Prefs tab of the EMC Settings control panel.

Conditions

Measurement view: EMC - EMI Display

Group

Display commands

Syntax

```
DISPlay:EMC:SHOW:LIMit:LEGend { 0 | OFF | 1 | ON}
DISPlay:EMC:SHOW:LIMit:LEGend?
```

Arguments

OFF or 0 hides the limit legends and label.

ON or 1 display the limit legends and label.

Returns

<Boolean> 0,1

Examples

DISPLAY:EMC:SHOW:LIMIT:LEGEND might return 0, indicating that the limit legends and label are hidden on the plot.

DISPlay:EMC:SHOW:SPOT:INDIcator

Sets or queries to display or hide the spots on the plot in EMC View.

Conditions

Measurement view: EMC - EMI Display

Group

Display commands

Syntax

```
DISPlay:EMC:SHOW:SPOT:INDIcator {0 | OFF | 1 | ON}
DISPlay:EMC:SHOW:SPOT:INDIcator?
```

Arguments

OFF or 0 hides the spots

ON or 1 display the spots

Returns

<Boolean> 0,1

Examples

DISPLAY:EMC:SHOW:SPOT:INDICATOR 1 displays the spots on the plot in EMC View.

DISPlay:EMC:SHOW:WIZArd

Sets or queries to display or hide the EMC wizard when EMCVu is launched by enabling or disabling the check box in the Prefs tab of the EMC Settings control panel.

Conditions

Measurement view: EMC - EMI Display

Group

Display commands

Syntax

```
DISPlay:EMC:SHOW:WIZArd {0 | OFF | 1 | ON}
DISPlay:EMC:SHOW:WIZArd?
```

Arguments

OFF or 0 hides the EMC wizard

ON or 1 displays the EMC wizard

Returns

<Boolean> 0,1

Examples

DISPLAY:EMC:SHOW:WIZARD ON displays the EMC wizard when EMCVu is launched.

DISPlay:EMC:SPOT<x>:SELect

Sets or queries to enable or disable the selected spot in emission results of EMC view.

<x> = <NR1> Indicates the spot number

Conditions

Measurement view: EMC - EMI Display

Group

Display commands

Syntax

```
DISPlay:EMC:SPOT<x>:SELect { 0 | 1 | ON | OFF }
DISPlay:EMC:SPOT<x>:SELect ?
```

Arguments

OFF or 0 disables the spot

1 or ON enables the spot

Returns

<Boolean> 0,1

Examples

`DISPlay:EMC:SPOT:SELect?` might return 1, indicating that spot 2 is enabled.

DISPlay:EMC:TRACe:LEGEnd

Sets or queries to display or hide the trace legend by enabling or disabling the check box in the Prefs tab of the EMC Settings control panel.

Conditions

Measurement view: EMC - EMI Display

Group

Display commands

Syntax

```
DISPlay:EMC:TRACe:LEGEnd { 0 | OFF | 1 | ON}
DISPlay:EMC:TRACe:LEGEnd?
```

Arguments

0 or OFF hides the Trace Legend

1 or ON display the Trace Legend

Returns

<Boolean> 0, 1

Examples

`DISPLAY:EMC:TRACE:LEGEND` returns `0`, indicating that the trace legend is hidden.

DISPlay:EMC:WINDow:TRACe:GRATicule:GRID:STATe

Sets or queries to display or hide the Graticule grid in the EMC view.

Conditions

Measurement view: EMC - EMI Display

Group

Display commands

Syntax

```
DISPlay:EMC:WINDow:TRACe:GRATicule:GRID:STATe { 0 | 1 | ON | OFF }
DISPlay:EMC:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or 0 hides the Graticule grid in the graph 1 or

`ON` display the Graticule grid in the graph

Returns

<Boolean> 0, 1

Examples

`DISPLAY:EMC:WINDOW:TRACE:GRATICULE:GRID:STATE OFF` hides the graticule grid in the EMC view.

DISPlay:EMC:X[:SCALe]:AUTO (No Query Form)

Sets the horizontal scale automatically to fit the waveform to the complete screen in EMC View.

Conditions

Measurement view: EMC - EMI Display

Group

Display commands

Syntax

```
DISPlay:EMC:X[:SCALe]:AUTO
```

Arguments

None

Returns

None

Examples

`DISPLAY:EMC:X[:SCALE]:AUTO` sets the horizontal scale automatically to fit the waveform to the complete screen.

DISPlay:EMC:X[:SCALe]:STARt

Sets and queries the Zoom Start frequency in EMC View.

Conditions

Measurement view: EMC - EMI Display

Group

Display commands

Syntax

```
DISPlay:EMC:X[:SCALe]:STARt <value>
DISPlay:EMC:X[:SCALe]:STARt?
```

Arguments

<value> = <NRf>

Returns

`<NRf>`

Examples

DISPLAY:EMC:X[:SCALE]:START might return 64.29, indicating that the zoom start frequency is 64.2 kHz.

DISPlay:EMC:X[:SCALe]:STOP

Sets and queries the Zoom stop frequency in EMC View.

Conditions

Measurement view: EMC - EMI Display

Group

Display commands

Syntax

```
DISPlay:EMC:X[:SCALe]:STOP <value>
DISPlay:EMC:X[:SCALe]:STOP?
```

Arguments

<value> = <NRf>

Returns

<NRf>

Examples

DISPLAY:EMC:X[:SCALE]:STOP might return 732.29, indicating that the zoom stop frequency is 732.2 kHz.

DISPlay:EMC:Y[:SCALe]

Sets or queries the vertical scale value in EMC View.

Conditions

Measurement view: EMC - EMI Display

Group

Display commands

Syntax

```
DISPlay:EMC:Y[:SCALe] <value>
DISPlay:EMC:Y[:SCALe]?
```

Arguments

<value> = <NRf>

Returns

<NRf>

Examples

DISPLAY:EMC:Y[:SCALE] 100 sets the vertical scale to 100 dB.

DISPlay:EMC:Y[:SCALe]:AUTO (No Query Form)

Sets the vertical scale automatically to fit the waveform to the complete screen in the EMC View.

Conditions

Measurement view: EMC - EMI Display

Group

Display commands

Syntax

```
DISPlay:EMC:Y[:SCALe]:AUTO
```

Arguments

None

Returns

None

Examples

`DISPLAY:EMC:Y[:SCALE]:AUTO` sets the vertical scale automatically to fit the waveform to the complete screen.

DISPlay:EMC:Y[:SCALe]:OFFSet

Sets or queries the vertical Position (offset) value in EMC View.

Conditions

Measurement view: EMC - EMI Display

Group

Display commands

Syntax

```
DISPlay:EMC:Y[:SCALe]:OFFSet <value>
DISPlay:EMC:Y[:SCALe]:OFFSet?
```

Arguments

`<value> = <NRf>`

Returns

`<NRf>`

Examples

`DISPLAY:EMC:Y[:SCALE]:OFFSET` might return `45.74`, indicating the vertical Position (offset) value is 45.7 dBuV.

DISPlay:EVM[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe

Sets or queries the graticule grid view state for the EVM view.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: EVM versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod EVM versus Time).

Group

Display commands

Syntax

```
DISPlay:EVM:MEASview<y>:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:EVM:MEASview<y>:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or 0 hides the graticule grid.

`ON` or 1 shows the graticule grid.

Returns

OFF or 0

ON or 1

Examples

`DISPlay:EVM:MEASview2:WINDow:TRACe:GRATicule:GRID:STATe ON` shows the graticule grid on the EVM Measview 2.

DISPlay:EVM[:MEASview<y>]:Y[:SCALe]

Sets or queries the vertical range of the EVM versus Time graph.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: EVM versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Constellation).

Group

Display commands

Syntax

```
DISPlay:EVM[:MEASview<y>]:Y[:SCALe]<value>
DISPlay:EVM[:MEASview<y>]:Y[:SCALe]?
```

Related Commands

DISPlay:EVM:[MEASview1]:Y[:SCALe]:OFFSet (on page 737)

Arguments

`<value>::<NRf>` specifies the vertical range.

Range: 1 to 100%.

Returns

`<NRf>`

Examples

`DISPlay:EVM:MEASview1:Y:SCALe 50` sets the vertical range to 50% in the EVM versus Time display Measurement view 1.

DISPlay:EVM[:MEASview<y>]:Y[:SCALe]:AUTO (No query form)

Sets the vertical scale automatically to fit the waveform to the screen in the EVM versus Time display.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: EVM versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod EVM versus Time).

Group

Display commands

Syntax

```
DISPlay:EVM[:MEASview<y>]:Y[:SCALe]:AUTO
```

Arguments

None

Examples

`DISPlay:EVM:MEASview2:Y:SCALe:AUTO` sets the vertical scale automatically to fit the waveform to the screen of EVM versus Time Measurement view 2.

DISPlay:EVM[:MEASview<y>]:Y[:SCALe]:OFFSet

Sets or queries the minimum vertical value (bottom edge) of the EVM versus Time graph.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: EVM versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Constellation).

Group

Display commands

Syntax

```
DISPlay:EVM[:MEASview<y>]:Y[:SCALe]:OFFSet <value>
DISPlay:EVM[:MEASview<y>]:Y[:SCALe]:OFFSet?
```

Related Commands

DISPlay:EVM[:MEASview<y>]:Y[:SCALe] (on page 736)

Arguments

<value>::<NRf> specifies the minimum vertical value.

Range: -100 to 100%.

Returns

<NRf>

Examples

DISPlay:EVM:MEASview2:Y:SCALe:OFFSet -9.5 sets the minimum vertical value to -9.5% in the EVM versus Time graph Measurement View 2.

DISPlay:FDVTime[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe

Sets or queries the graticule grid view state for the FDV view.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Frequency deviation versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Frequency deviation versus Time).

Group

Display commands

Syntax

```
DISPlay:FDVTime:MEASview<y>:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:FDVTime:MEASview<y>:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

OFF or 0 hides the graticule grid.

ON or 1 shows the graticule grid.

Returns

OFF

ON

Examples

DISPlay:FDVTime:MEASview2:WINDow:TRACe:GRATicule:GRID:STATe ON shows the graticule grid on the Frequency deviation versus Time Measview 2.

DISPlay:FDVTime[:MEASview<y>]:Y[:SCALe]

Sets or queries the vertical range of the Frequency deviation versus Time graph.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Frequency deviation versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Frequency deviation versus Time).

Group

Display commands

Syntax

```
DISPlay:FDVTime:[MEASview<y>]:Y[:SCALe]<value>
DISPlay:FDVTime:[MEASview<y>]:Y[:SCALe]?
```

Related Commands

DISPlay:FDVTime[:MEASview<y>]:Y[:SCALe]:OFFSet (on page 740)

Arguments

`<value>::<NRf>` specifies the vertical range.

Range: 10 Hz to 120 MHz.

Returns

`<NRf>`

Examples

`DISPlay:FDVTime:MEASview1:Y:SCALe 30 MHz` sets the vertical range to 30 MHz in the Frequency deviation versus Time display Measurement view 1.

DISPlay:FDVTime[:MEASview<y>]:Y[:SCALe]:AUTO (No query form)

Sets the vertical scale automatically to fit the waveform to the screen in the Frequency deviation versus Time display.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Frequency deviation versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Frequency deviation versus Time).

Group

Display commands

Syntax

```
DISPlay:FDVTime:MEASview<y>:Y[:SCALe]:AUTO
```

Arguments

None

Examples

`DISPlay:FDVTime:MEASview2:Y:SCALe:AUTO` sets the vertical scale automatically to fit the waveform to the screen of Frequency deviation versus Time Measurement view 2.

DISPlay:FDVTime[:MEASview<y>]:Y[:SCALe]:OFFSet

Sets or queries the minimum vertical value (bottom edge) of the Frequency deviation versus Time graph.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Frequency deviation versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Frequency deviation versus Time).

Group

Display commands

Syntax

```
DISPlay:FDVTime[:MEASview<y>]:Y[:SCALe]:OFFSet<value>
DISPlay:FDVTime[:MEASview<y>]:Y[:SCALe]:OFFSet?
```

Related Commands

DISPlay:FDVTime[:MEASview<y>]:Y[:SCALe] (on page 738)

Arguments

`<value>::<NRf>` specifies the vertical offset.

Range: -60 MHz to +60 MHz.

Returns

`<NRf>`

Examples

`DISPlay:FDVTime:MEASview2:Y:SCALe:OFFSet -14.5 MHz` sets the minimum vertical value to -14.5 MHz in the Frequency deviation versus Time graph Measurement View 2.

DISPlay:{FSETtling|PSETtling}:MARKer:SHOW:STATe

Determines whether to show or hide the readout for the selected marker in the Frequency or Phase settling views.

Conditions

Measurement views: Frequency and Phase Settling

Group

Display commands

Syntax

```
DISPlay:{FSETtling|PSETtling}:MARKer:SHOW:STATe { OFF | ON | 0 | 1 }
DISPlay:{FSETtling|PSETtling}:MARKer:SHOW:STATe?
```

Arguments

`OFF` or 0 hides the readout for the selected marker in the view.

`ON` or 1 shows the readout for the selected marker in the view.

Returns

See Arguments.

Examples

`DISPLAY:FSETTLING:MARKER:SHOW:STATE ON` shows the readout for the selected marker in the view.

DISPlay:{FSETtling|PSETtling}:TIME:DECimal

Sets or queries the number of values to the right of the decimal point to include in the Settling Time and Settling Time from Trigger results.

Conditions

Measurement views: Frequency and Phase Settling Time

Group

Display commands

Syntax

```
DISPlay:{FSETtling|PSETtling}:TIME:DECimal <value>
DISPlay:{FSETtling|PSETtling}:TIME:DECimal?
```

Arguments

`<value>::<NR1>` specifies the number digits to the right of the decimal point to include in Settling Time and Settling Time from Trigger results. Range: 0 to 6.

Returns

`<NR1>`

Examples

`DISPLAY:PSETTLING:TIME:DECIMAL 3` sets the number of values to the right of the decimal point to 3 for the Phase Settling time measurement.

DISPlay:{FSETtling|PSETtling}:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions

Measurement views: Frequency and Phase Settling

Group

Display commands

Syntax

```
DISPlay:{FSETtling|PSETtling}:WINDow:TRACe:GRATicule:GRID: STATe { OFF | ON | 0 | 1 }
DISPlay:{FSETtling|PSETtling}:WINDow:TRACe:GRATicule:GRID: STATe?
```

Arguments

`OFF` or 0 hides the graticule grid.

`ON` or 1 shows the graticule grid.

Returns

See Arguments.

Examples

`DISPLAY:FSETTLING:WINDOW:TRACE:GRATICULE:GRID:STATE ON` shows the graticule grid on the screen for the Frequency Settling display.

DISPlay:{FSETtling|PSETtling}:X[:SCALe]

Sets or queries the horizontal scale (full-scale time) of the Settling Time graph. Programming a specified scale sets `DISPlay:{FSETtling|PSETtling}:X[:SCALe] AUTO:STATe OFF`.

Conditions

Measurement views: Frequency versus Time

Group

Display commands

Syntax

```
DISPlay:{FSETtling|PSETtling}:X[:SCALe] <value>
DISPlay:{FSETtling|PSETtling}:X[:SCALe]?
```

Related Commands

DISPlay:{FSETtling|PSETtling}:X[:SCALe]:AUTO:STATe (on page 743)

DISPlay:{FSETtling|PSETtling}:X[:SCALe]:OFFSet (on page 744)

Arguments

`<value>::={ <NRf> | MAXimum | MINimum }` specifies the horizontal scale in full-scale time. MAXimum and MINimum represent the upper and lower limits of the setting range, respectively.

Use the DISPlay:{FSETtling|PSETtling}:X[:SCALe]:MAXimum? (on page 743) and DISPlay:{FSETtling|PSETtling}:X[:SCALe]:MINimum? (on page 744) queries to get the upper and lower limit values of the setting range.

Returns

See Arguments.

Examples

`DISPLAY:FSETTLING:X:SCALE 25.6us` sets the horizontal scale to 25.6 µs.

DISPlay:{FSETtling|PSETtling}:X[:SCALe]:AUTO (No Query Form)

Sets the horizontal scale automatically to fit the waveform to the screen in the Frequency and Phase Settling views. Executing this command sets `DISPlay:{FSETtling|PSETtling}:X[:SCALe]:AUTO:STATe ON`.

Conditions

Measurement views: Frequency and Phase Settling

Group

Display commands

Syntax

```
DISPlay:{FSETtling|PSETtling}:X[:SCALe]:AUTO
```

Related Commands

DISPlay:{FSETtling|PSETtling}:X[:SCALe]:AUTO:STATe (on page 743)

Arguments

None

Examples

`DISPLAY:FSETTLING:X:SCALE:AUTO` sets the horizontal scale automatically to fit the waveform to the screen.

DISPlay:{FSETtling|PSETtling}:X[:SCALe]:AUTO:STATe

Determines whether to set the horizontal scale automatically or manually.

Conditions

Measurement views: Frequency and Phase Settling

Group

Display commands

Syntax

```
DISPlay:{FSETtling|PSETtling}:X[:SCALe]:AUTO:STATe { OFF | ON | 0 | 1 }
DISPlay:{FSETtling|PSETtling}:X[:SCALe]:AUTO:STATe?
```

Arguments

`OFF` or 0 specifies that the horizontal scale is set manually. To set it, use the DISPlay:{FSETtling|PSETtling}:X[:SCALe] (on page 742) and DISPlay:{FSETtling|PSETtling}:X[:SCALe]:OFFSet (on page 744) commands.

`ON` or 1 specifies that the horizontal scale is set automatically.

Returns

See Arguments.

Examples

`DISPLAY:FSETTLING:X:SCALE:AUTO:STATE ON` specifies that the horizontal scale is set automatically.

DISPlay:{FSETtling|PSETtling}:X[:SCALe]:MAXimum? (Query Only)

Queries the upper limit of the horizontal scale setting range.

Conditions

Measurement views: Frequency and Phase Settling

Group

Display commands

Syntax

```
DISPlay:{FSETtling|PSETtling}:X[:SCALe]:MAXimum?
```

Related Commands

DISPlay:{FSETtling|PSETtling}:X[:SCALe] (on page 742)

Returns

<NRf> The upper limit of the horizontal scale setting range.

Examples

DISPLAY:FSETTLING:X:SCALE:MAXIMUM? might return 18.135E-3, indicating that the upper limit of the horizontal scale setting range is 18.135 ms.

DISPlay:{FSETtling|PSETtling}:X[:SCALe]:MINimum? (Query Only)

Queries the lower limit of the horizontal scale setting range.

Conditions

Measurement views: Frequency and Phase Settling

Group

Display commands

Syntax

DISPlay:{FSETtling|PSETtling}:X[:SCALe]:MINimum?

Related Commands

DISPlay:FVTime[:MEASview<y>]:X[:SCALe] (on page 749)

Returns

<NRf> The lower limit of the horizontal scale setting range.

Examples

DISPLAY:FSETTLING:X:SCALE:MINIMUM? might return 10.0E-9, indicating that the lower limit of the horizontal scale setting range is 10.0 ns.

DISPlay:{FSETtling|PSETtling}:X[:SCALe]:OFFSet

Sets or queries the minimum horizontal value (left edge) of the Frequency versus Time graph. Programming a specified offset sets DISPlay:{FSETtling|PSETtling}:X[:SCALe] AUTO:STATe OFF.

Conditions

Measurement views: Frequency and Phase Settling

Group

Display commands

Syntax

DISPlay:{FSETtling|PSETtling}:X[:SCALe]:OFFSet <value>
DISPlay:{FSETtling|PSETtling}:X[:SCALe]:OFFSet?

Related Commands

DISPlay:{FSETtling|PSETtling}:X[:SCALe]:AUTO:STATe (on page 743)

DISPlay:{FSETtling|PSETtling}:X[:SCALe] (on page 742)

Arguments

`<value>::={ <NRf> | MAXimum | MINimum }` specifies the horizontal offset. MAXimum and MINimum represent the upper and lower limits of the setting range, respectively.

Use the DISPlay:{FSETtling|PSETtling}:X[:SCALe]:OFFSet:MAXimum? (on page 743) and DISPlay:{FSETtling|PSETtling}:X[:SCALe]:OFFSet:MINimum? (on page 744) queries to get the upper and lower limit values of the setting range.

Returns

See Arguments.

Examples

`DISPLAY:FSETTLING:X:SCALE:OFFSET 800ns` sets the minimum horizontal value to 800 ns in the Frequency Settling graph.

DISPlay:{FSETtling|PSETtling}:X[:SCALe]:OFFSet:MAXimum? (Query Only)

Queries the upper limit of the horizontal offset setting range.

Conditions

Measurement views: Frequency and Phase Settling

Group

Display commands

Syntax

`DISPlay:{FSETtling|PSETtling}:X[:SCALe]:OFFSet:MAXimum?`

Related Commands

DISPlay:{FSETtling|PSETtling}:X[:SCALe]:OFFSet (on page 744)

Returns

`<NRf>` The upper limit of the horizontal offset setting range.

Examples

`DISPLAY:FSETTLING:X:SCALE:OFFSET:MAXIMUM?` might return `338.186000E-6`, indicating that the upper limit of the horizontal offset setting range is 338.186 µs.

DISPlay:{FSETtling|PSETtling}:X[:SCALe]:OFFSet:MINimum? (Query Only)

Queries the lower limit of the horizontal offset setting range.

Conditions

Measurement views: Frequency and Phase Settling

Group

Display commands

Syntax

```
DISPlay:{FSETtling|PSETtling}:X[:SCALe]:OFFSet:MINimum?
```

Related Commands

DISPlay:FVTime:X[:SCALe]:OFFSet (on page 752)

Returns

<NRf> The lower limit of the horizontal offset setting range.

Examples

DISPLAY:FSETTLING:X:SCALE:OFFSET:MINIMUM? might return –338.186000E–6, indicating that the lower limit of the horizontal offset setting range is -338.186 µs.

DISPlay:{FSETtling|PSETtling}:Y[:SCALe]

Sets or queries the vertical range of the Frequency versus Time graph.

Conditions

Measurement views: Frequency and Phase Settling

Group

Display commands

Syntax

```
DISPlay:{FSETtling|PSETtling}:Y[:SCALe] <value>
DISPlay:{FSETtling|PSETtling}:Y[:SCALe]?
```

Related Commands

DISPlay:{FSETtling|PSETtling}:Y[:SCALe]:OFFSet (on page 747)

Arguments

<value>::<NRf> specifies the vertical range. Range: 10 Hz to 120 MHz.

Returns

<NRf>

Examples

DISPlay:FSETTLING:Y:SCALe 30E6 sets the vertical range to 30 MHz in the Frequency Settling graph.

DISPlay:{FSETtling|PSETtling}:Y[:SCALe]:AUTO (No Query Form)

Sets the vertical scale automatically to fit the waveform to the screen in the Frequency and Phase Settling views.

Conditions

Measurement views: Frequency and Phase Settling

Group

Display commands

Syntax

```
DISPlay:{FSETtling|PSETtling}:Y[:SCALe]:AUTO
```

Arguments

None

Examples

`DISPLAY:FSETTLING:Y:SCALE:AUTO` sets the vertical scale automatically to fit the waveform to the screen.

DISPlay:{FSETtling|PSETtling}:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (the value at the center of the vertical axis) in the Frequency and Phase Settling graphs.

Conditions

Measurement views: Frequency and Phase Settling

Group

Display commands

Syntax

```
DISPlay:{FSETtling|PSETtling}:Y[:SCALe]:OFFSet <value>
DISPlay:{FSETtling|PSETtling}:Y[:SCALe]:OFFSet?
```

Related Commands

DISPlay:{FSETtling|PSETtling}:Y[:SCALe] (on page 746)

Arguments

`<value>::<NRf>` specifies the vertical offset. Range: -60 MHz to +60 MHz.

Returns

`<NRf>`

Examples

`DISPLAY:FVTIME:Y:SCALE:OFFSET -14.5E6` sets the vertical offset to -14.5 MHz in the Frequency Settling graph.

DISPlay:{FSETtling|PSETtling}:Y[:SCALe]:PDIVision

Sets or queries the vertical scale (per division) of the Frequency and Phase Settling graphs.

Conditions

Measurement views: Frequency and Phase Settling

Group

Display commands

Syntax

```
DISPlay:{FSETtling|PSETtling}:Y[:SCALe]:PDIVision <value>
DISPlay:{FSETtling|PSETtling}:Y[:SCALe]:PDIVision?
```

Arguments

<value>::<NRf> specifies the vertical scale (per division).

Range:

- Frequency Settling (Hz/div) : 500 to the Maximum bandwidth of instrument
- Phase Settling (degrees/div): 0.10 to X.

Returns

<NRf>

Examples

DISPLAY:PSETTLING:Y:SCALE:PDIVISION 2.7 sets the vertical scale to 27 degrees.

DISPlay:FVTime[:MEASview<y>]:READout[:STATe]

Determines whether to show or hide the measurement parameter readouts in the Frequency versus Time view.

Conditions

Measurement views: Frequency versus Time

Group

Display commands

Syntax

DISPlay:FVTime[:MEASview<y>]:READout[:STATe] { OFF | ON | 0 | 1 }

DISPlay:FVTime[:MEASview<y>]:READout[:STATe]?

Arguments

OFF or 0 hides the parameter readouts.

ON or 1 shows the parameter readouts.

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. The default value is 1

Examples

DISPLAY:FVTIME:MEASVIEW1:READOUT:STATE ON shows the parameter readouts on the screen.

DISPlay:FVTime[:MEASview<y>]:SELected:TIMestamp

Sets or queries the display of the time stamp in the ogram plot.

Conditions

Applicable measurement view must be active.

Group

Display commands

Syntax

DISPlay:FVTime[:MEASview<y>]:SELected:TIMestamp {OFF | ON | 0 | 1}

DISPlay:FVTime[:MEASview<y>]:SELected:TIMestamp?

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional. By default y is 1.

`OFF` or `0` removes or turns off the time stamp.

`ON` or `1` displays or turns on the time stamp.

Returns

`{OFF | ON | 0 | 1}`

Examples

`DISPlay:FVTime:MEASview1:SELected:TIMestamp?` queries the display status of the time stamp in the ogram plot view 1.

DISPlay:FVTime[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions

Measurement views: Frequency versus Time

Group

Display commands

Syntax

```
DISPlay:FVTime[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:FVTime[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

`OFF` or 0 hides the graticule grid.

`ON` or 1 shows the graticule grid.

Returns

See Arguments.

Examples

`DISPLAY:FVTIME:MEASview1:WINDOW:TRACE:GRATICULE:GRID:STATE ON` shows the graticule grid on the Frequency versus Time view.

DISPlay:FVTime[:MEASview<y>]:X[:SCALe]

Sets or queries the horizontal scale (full-scale time) of the Frequency versus Time graph. Programming a specified scale sets `DISPlay:FVTime[:MEASview<y>]:X[:SCALe]AUTO:STATe OFF`.

Conditions

Measurement views: Frequency versus Time

Group

Display commands

Syntax

```
DISPlay:FVTime[:MEASview<y>]:X[:SCALe] <value>
DISPlay:FVTime[:MEASview<y>]:X[:SCALe]?
```

Related Commands

DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:AUTO:STATe (on page 751)

DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:OFFSet (on page 752)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<value>::={ <NRf> | MAXimum | MINimum } specifies the horizontal scale in full-scale time. MAXimum and MINimum represent the upper and lower limits of the setting range, respectively.

Use the DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:MAXimum? (on page 751) and DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:MINimum? (on page 752) queries to get the upper and lower limit values of the setting range.

Returns

See Arguments.

Examples

`DISPLAY:FVTIME:MEASview1:X:SCALE 25.6us` sets the horizontal scale to 25.6 µs.

DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:AUTO (No Query Form)

Sets the horizontal scale automatically to fit the waveform to the screen in the Frequency versus Time view. Executing this command sets `DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:AUTO:STATe ON`.

Conditions

Measurement views: Frequency versus Time

Group

Display commands

Syntax

```
DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:AUTO
```

Related Commands

DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:AUTO:STATe (on page 751)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Examples

`DISPLAY:FVTIME:MEASview1:X:SCALE:AUTO` sets the horizontal scale automatically to fit the waveform to the screen.

DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:AUTO:STATe

Determines whether to set the horizontal scale automatically or manually.

Conditions

Measurement views: Frequency versus Time

Group

Display commands

Syntax

```
DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:AUTO:STATe { OFF | ON | 0 | 1 }
DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:AUTO:STATe?
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

OFF or 0 specifies that the horizontal scale is set manually. To set it, use the DISPlay:FVTime[:MEASview<y>]:X[:SCALe] (on page 749) and DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:OFFSet (on page 752) commands.

ON or 1 specifies that the horizontal scale is set automatically.

Returns

See Arguments.

Examples

DISPLAY:FVTIME:MEASview1:X:SCALE:AUTO:STATE ON specifies that the horizontal scale is set automatically.

DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:MAXimum? (Query Only)

Queries the upper limit of the horizontal scale setting range.

Conditions

Measurement views: Frequency versus Time

Group

Display commands

Syntax

```
DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:MAXimum?
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Related Commands

DISPlay:FVTime[:MEASview<y>]:X[:SCALe] (on page 749)

Returns

<NRf> The upper limit of the horizontal scale setting range.

Examples

DISPLAY:FVTIME:MEASview1:X:SCALE:MAXIMUM? might return 18.135E-3, indicating that the upper limit of the horizontal scale setting range is 18.135 ms.

DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:MINimum? (Query Only)

Queries the lower limit of the horizontal scale setting range.

Conditions

Measurement views: Frequency versus Time

Group

Display commands

Syntax

DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:MINimum?

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Related Commands

DISPlay:FVTime[:MEASview<y>]:X[:SCALe] (on page 749)

Returns

<NRf> The lower limit of the horizontal scale setting range.

Examples

DISPLAY:FVTIME:MEASview1:X:SCALE:MINIMUM? might return 10.0E-9, indicating that the lower limit of the horizontal scale setting range is 10.0 ns.

DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:OFFSet

Sets or queries the minimum horizontal value (left edge) of the Frequency versus Time graph. Programming a specified offset sets DISPlay:FVTime[:MEASview<y>]:X[:SCALe]AUTO:STATe OFF.

Conditions

Measurement views: Frequency versus Time

Group

Display commands

Syntax

DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:OFFSet <value>
DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:OFFSet?

Related Commands

DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:AUTO:STATe (on page 751)

DISPlay:FVTime[:MEASview<y>]:X[:SCALe] (on page 749)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<value>::={ <NRf> | MAXimum | MINimum } specifies the horizontal offset. MAXimum and MINimum represent the upper and lower limits of the setting range, respectively.

Use the DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MAXimum? (on page 753) and DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MINimum? (on page 754) queries to get the upper and lower limit values of the setting range.

Returns

See Arguments.

Examples

DISPLAY:FVTIME:MEASview1:X:SCALE:OFFSET 800ns sets the minimum horizontal value to 800 ns in the Frequency versus Time graph.

DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MAXimum? (Query Only)

Queries the upper limit of the horizontal offset setting range.

Conditions

Measurement views: Frequency versus Time

Group

Display commands

Syntax

DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MAXimum?

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Related Commands

DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:OFFSet (on page 752)

Returns

<NRf> The upper limit of the horizontal offset setting range.

Examples

DISPLAY:FVTIME:MEASview1:X:SCALE:OFFSET:MAXIMUM? might return -1.812E-3, indicating that the upper limit of the horizontal offset setting range is -1.812 ms.

DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MINimum? (Query Only)

Queries the lower limit of the horizontal offset setting range.

Conditions

Measurement views: Frequency versus Time

Group

Display commands

Syntax

```
DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MINimum?
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Related Commands

DISPlay:FVTime[:MEASview<y>]:X[:SCALe]:OFFSet (on page 755)

Returns

<NRf> The lower limit of the horizontal offset setting range.

Examples

`DISPLAY:FVTIME:MEASview1:X:SCALE:OFFSET:MINIMUM?` might return `-16.28E-3`, indicating that the lower limit of the horizontal offset setting range is -16.28 ms.

DISPlay:FVTime[:MEASview<y>]:Y[:SCALe]

Sets or queries the vertical range of the Frequency versus Time graph.

Conditions

Measurement views: Frequency versus Time

Group

Display commands

Syntax

```
DISPlay:FVTime[:MEASview<y>]:Y[:SCALe] <value>
DISPlay:FVTime[:MEASview<y>]:Y[:SCALe]?
```

Related Commands

DISPlay:FVTime[:MEASview<y>]:Y[:SCALe]:OFFSet (on page 755)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<value>::<NRf> specifies the vertical range. Range: 10 Hz to 120 MHz.

Returns

<NRf>

Examples

DISPlay:FVTime:MEASview1:Y:SCALe 30MHz sets the vertical range to 30 MHz in the Frequency versus Time graph.

DISPlay:FVTime[:MEASview<y>]:Y[:SCALe]:AUTO (No Query Form)

Sets the vertical scale automatically to fit the waveform to the screen in the Frequency versus Time view.

Conditions

Measurement views: Frequency versus Time

Group

Display commands

Syntax

DISPlay:FVTime[:MEASview<y>]:Y[:SCALe]:AUTO

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Examples

DISPLAY:FVTIME:MEASview1:Y:SCALE:AUTO sets the vertical scale automatically to fit the waveform to the screen.

DISPlay:FVTime[:MEASview<y>]:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (the value at the center of the vertical axis) in the Frequency versus Time graph.

Conditions

Measurement views: Frequency versus Time

Group

Display commands

Syntax

DISPlay:FVTime[:MEASview<y>]:Y[:SCALe]:OFFSet <value>
DISPlay:FVTime[:MEASview<y>]:Y[:SCALe]:OFFSet?

Related Commands

DISPlay:FVTime[:MEASview<y>]:Y[:SCALe] (on page 754)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<value>::<NRf> specifies the vertical offset. Range: -60 MHz to +60 MHz.

Returns

<NRf>

Examples

DISPLAY:FVTIME:MEASview1:Y:SCALE:OFFSET -14.5MHz sets the vertical offset to −14.5 MHz in the Frequency versus Time graph.

DISPlay:GENeral:MEASview<y>:DELete (No Query Form)

Deletes a measurement view in the general signal viewing.

Conditions

Measurement views: General signal viewing

Group

Display commands

Syntax

```
DISPlay:GENeral:MEASview<y>:DELete { SPECtrum | DPX | AVTime| FVTime | PHVTime | IQVTime | SGRam |
    TOVerview }
```

Arguments

The following table shows the view arguments. The arguments are the string type.

General signal viewing views

Argument	View
SPECtrum	Spectrum
DPX	DPX (Digital Phosphor) spectrum
AVTime	Amplitude versus Time
FVTime	Frequency versus Time
PHVTime	Phase versus Time
IQVTime	IQ versus Time
SGRam	Spectrogram
TOVerview	Time overview

If you attempt to delete a view that is not displayed on screen, the error (-200, "Execution error; Measurement not running") will be returned.

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Examples

DISPLAY:GENERAL:MEASVIEW1:DELETE DPX deletes the DPX spectrum view with Measurement View 1.

DISPlay:GENeral:MEASview<y>[:SOURce<x>]:NEW (No Query Form)

Displays a new measurement view in the general signal viewing.

Conditions

Measurement views: General signal viewing with specified Measurement view number.

Group

Display commands

Syntax

```
DISPlay:GENeral:MEASview<y>[:SOURce<x>]:NEW { SPECtrum | DPX | AVTime| FVTime | PHVTime | IQVTime |
    SGRam | TOVerview }
```

Arguments

The following table shows the view arguments. The arguments are the string type.

General signal viewing views

Argument	View
SPECtrum	Spectrum
DPX	DPX (Digital Phosphor) spectrum
AVTime	Amplitude versus Time
FVTime	Frequency versus Time
PHVTime	Phase versus Time
IQVTime	IQ versus Time
SGRam	Spectrogram
TOVerview	Time overview

If you attempt to open a view that is currently displayed on screen, the error (-200, "Execution error; Measurement is already running") will be returned.

<x> = 1 to 8, representing Range of Source.

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Examples

`DISPlay:GENeral:MEASview1:SOURce1:NEW DPX` creates the DPX spectrum view for source1.

DISPlay:GENeral:MEASview<y>:SELect

Selects a measurement view in the general signal viewing on the screen. The query command returns the currently selected view.

Selecting a measurement optimizes it. Other measurements may be optimized as a side effect. Refer to the `DISPlay:WINDow:OPTimized:MEASurement?` query.

Conditions

Measurement views: General signal viewing

Group

Display commands

Syntax

```
DISPlay:GENeral:MEASview<y>:SELect { SPECtrum | DPX | AVTime| FVTime | PHVTime | IQVTime | SGRam |
    TOVerview }
DISPlay:GENeral:MEASview<y>:SELect?
```

Arguments

The following table shows the view arguments. The arguments are the string type.

General signal viewing views

Argument	View
SPECtrum	Spectrum
DPX	DPX (Digital Phosphor) spectrum
AVTime	Amplitude versus Time
FVTime	Frequency versus Time
PHVTime	Phase versus Time
IQVTime	IQ versus Time
SGRam	Spectrogram
TOVerview	Time overview

If you attempt to select a view that is not displayed on screen, the error (-200, "Execution error; Measurement not running") will be returned.

The time overview (TOVerview) cannot be selected as the primary measurement. If you select it, the error (-200, "Execution error; Time Overview cannot be Primary measurement") will be returned. If you use the DISPlay:GENeral MEASview:SELect? query with the time overview as the only measurement active, the error (-200, "Execution error; Analysis selected is not running") will be returned.

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

See Arguments.

Examples

`DISPLAY:GENERAL:MEASVIEW1:SELECT DPX` selects the DPX spectrum view with Measurement view 1.

DISPlay:GPRF:MEASview:DELete (No Query Form)

Deletes a selected measurement view in the RF measurements.

Conditions

Measurement views: RF measurements

Group

Display commands

Syntax

```
DISPlay:GPRF:MEASview:DELete { AM | FM | PM | CCDF | ACPower | FSETtling | MCPower | OBW | PNOise |
    PSETtling | SPURious |SEM }
```

Arguments

`AM` deletes the AM view.

`FM` deletes the FM view.

`PM` deletes the PM view.

`CCDF` deletes the CCDF view.

`ACPower` deletes the Channel power and ACPR view.

`FSETtling` deletes the Frequency Settling view.

`MCPower` deletes the MCPR view.

`OBW` deletes the Occupied Bandwidth view.

`PNOise` deletes the Phase Noise view (Option 11 only).

`PSETtling` deletes the Phase Settling view.

`SPURious` deletes the Spurious view.

`SEM` deletes the Spectral Emissions Mask view.

If you attempt to delete a view that is not displayed on screen, the error (-200, "Execution error; Measurement not running") will be returned.

Examples

`DISPlay:GPRF:MEASview:DELete ACPower` deletes the Channel power and ACPR view.

DISPlay:GPRF:MEASview:NEW (No Query Form)

Displays a new measurement view in the RF measurements.

Conditions

Measurement views: RF measurements

Group

Display commands

Syntax

```
DISPlay:GPRF:MEASview:NEW { AM | FM | PM | CCDF | ACPower | FSETtling MCPower | OBW | PNOise |
    PSETtling | SPURious |SEM }
```

Arguments

`AM` opens the AM view.

`FM` opens the AM view.

PM opens the AM view.

CCDF opens the CCDF view.

ACPower opens the Channel power and ACPR view.

FSETtling opens the Frequency Settling view.

MCPower opens the MCPR view.

OBW opens the Occupied Bandwidth view.

PNOise opens the Phase Noise view (Option 11 only).

PSETtling opens the Phase Settling view.

SPURious opens the Spurious view.

SEM opens the Spectral Emissions Mask view.

If you attempt to open a view that is currently displayed on screen, the error (-200, "Execution error; Measurement is already running") will be returned.

Examples

DISPlay:GPRF:MEASview:NEW ACPower creates the Channel power and ACPR view.

DISPlay:GPRF:MEASview:SELect

Selects a measurement view in the RF measurements on the screen. The query command returns the currently selected view.

Selecting a measurement optimizes it. Other measurements may be optimized as a side effect. Refer to the DISPlay:WINDow:OPTimized:MEASurement? query.

Conditions

Measurement views: RF measurements

Group

Display commands

Syntax

```
DISPlay:GPRF:MEASview:SELect { AM | FM | PM | CCDF | ACPower | FSETtling | MCPower | OBW | PNOise |
    PSETtling | SPURious |SEM }
DISPlay:GPRF:MEASview:SELect?
```

Arguments

AM selects the AM view.

FM selects the FM view.

PM selects the PM view.

CCDF selects the CCDF view.

ACPower selects the Channel power and ACPR view.

FSETtling selects the Frequency Settling view.

MCPower selects the MCPR view.

OBW selects the Occupied Bandwidth view.

PNOise selects the Phase Noise view (Option 11 only).

PSETtling selects the Phase Settling view.

SPURious selects the Spurious view.

SEM selects the Spectral Emissions Mask view.

If you attempt to select a view that is not displayed on screen, the error (-200, "Execution error; Measurement not running") will be returned.

Returns

See Arguments.

Examples

DISPlay:GPRF:MEASview:SELect ACPower selects the Channel power and ACPR view.

DISPlay:IQVTime[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions

Measurement views: RF I&Q versus Time

Group

Display commands

Syntax

```
DISPlay:IQVTime[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:IQVTime[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

OFF or 0 hides the graticule grid.

ON or 1 shows the graticule grid.

Returns

See Arguments.

Examples

DISPLAY:IQVTIME:MEASview1:WINDOW:TRACE:GRATICULE:GRID:STATE ON shows the graticule grid on the RF I&Q versus Time view.

DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]

Sets or queries the horizontal scale (full-scale time) of the RF I&Q versus Time graph. Programming a specified scale sets DISPlay:IQVTime[:MEASview<y>]:X[:SCALe] AUTO:STATe OFF.

Conditions

Measurement views: RF I&Q versus Time

Group

Display commands

Syntax

```
DISPlay:IQVTime[:MEASview<y>]:X[:SCALe] <value>
DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]?
```

Related Commands

DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:AUTO:STATe (on page 763)

DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:OFFSet (on page 764)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<value>::={ <NRf> | MAXimum | MINimum } specifies the horizontal scale in full-scale time. MAXimum and MINimum represent the upper and lower limits of the setting range, respectively. Use the DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:MAXimum? (on page 763) and DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:MINimum? (on page 764) queries to get the upper and lower limit values of the setting range.

Returns

See Arguments.

Examples

DISPLAY:IQVTIME:MEASview<y>:X:SCALE 100us sets the horizontal scale to 100 µs.

DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:AUTO (No Query Form)

Sets the horizontal scale automatically to fit the waveform to the screen in the RF I&Q versus Time view. Executing this command sets DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:AUTO:STATe ON.

Conditions

Measurement views: RF I&Q versus Time

Group

Display commands

Syntax

```
DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:AUTO
```

Related Commands

DISPlay:IQVTime[:MEASview<y>]:X[:SCALe] (on page 761)

DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:AUTO:STATe (on page 763)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Examples

DISPLAY:IQVTIME:MEASview1:X:SCALE:AUTO sets the horizontal scale automatically to fit the waveform to the screen.

DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:AUTO:STATe

Determines whether to set the horizontal scale automatically or manually.

Conditions

Measurement views: RF I&Q versus Time

Group

Display commands

Syntax

```
DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:AUTO:STATe { OFF | ON | 0 | 1 }
DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:AUTO:STATe?
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

OFF or 0 specifies that the horizontal scale is set manually. To set it, use the DISPlay:IQVTime[:MEASview<y>]:X[:SCALe] (on page 761) and DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:OFFSet (on page 764) command.

ON or 1 specifies that the horizontal scale is set automatically.

Returns

See Arguments.

Examples

DISPLAY:IQVTIME:MEASview1:X:SCALE:AUTO:STATE ON specifies that the horizontal scale is set automatically.

DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:MAXimum? (Query Only)

Queries the upper limit of the horizontal scale setting range.

Conditions

Measurement views: RF I&Q versus Time

Group

Display commands

Syntax

```
DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:MAXimum?
```

Related Commands

DISPlay:IQVTime[:MEASview<y>]:X[:SCALe] (on page 761)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

<NRf> The upper limit of the horizontal scale setting range.

Examples

DISPLAY:IQVTIME:MEASview1:X:SCALE:MAXIMUM? might return 18.135E-3, indicating that the upper limit of the horizontal scale setting range is 18.135 ms.

DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:MINimum? (Query Only)

Queries the lower limit of the horizontal scale setting range.

Conditions

Measurement views: RF I&Q versus Time

Group

Display commands

Syntax

DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:MINimum?

Related Commands

DISPlay:IQVTime[:MEASview<y>]:X[:SCALe] (on page 761)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

<NRf> The lower limit of the horizontal scale setting range.

Examples

DISPLAY:IQVTIME:MEASview1:X:SCALE:MINIMUM? might return 10.0E-9, indicating that the lower limit of the horizontal scale setting range is 10.0 ns.

DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:OFFSet

Sets or queries the minimum horizontal value (left edge) of the RF I&Q versus Time graph. Programming a specified offset sets DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]AUTO:STATe OFF.

Conditions

Measurement views: RF I&Q versus Time

Group

Display commands

Syntax

DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:OFFSet <value>
DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:OFFSet?

Related Commands

DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:AUTO:STATe (on page 763)

DISPlay:IQVTime[:MEASview<y>]:X[:SCALe] (on page 761)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<value>::={ <NRf> | MAXimum | MINimum } specifies the horizontal offset. MAXimum and MINimum represent the upper and lower limits of the setting range, respectively.

Use the DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MAXimum? (on page 765) and DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MINimum? (on page 766) queries to get the upper and lower limit values of the setting range.

Returns

See Arguments.

Examples

DISPLAY:IQVTIME:MEASview1:X:SCALE:OFFSET 800ns sets the minimum horizontal value to 800 ns in the RF I&Q versus Time graph.

DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MAXimum? (Query Only)

Queries the upper limit of the horizontal offset setting range.

Conditions

Measurement views: RF I&Q versus Time

Group

Display commands

Syntax

DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MAXimum?

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Related Commands

DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:OFFSet (on page 764)

Returns

<NRf> The upper limit of the horizontal offset setting range.

Examples

DISPLAY:IQVTIME:MEASview1:X:SCALE:OFFSET:MAXIMUM? might return -1.812E-3, indicating that the upper limit of the horizontal offset setting range is -1.812 ms.

DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MINimum? (Query Only)

Queries the lower limit of the horizontal offset setting range.

Conditions

Measurement views: RF I&Q versus Time

Group

Display commands

Syntax

```
DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MINimum?
```

Related Commands

DISPlay:IQVTime[:MEASview<y>]:X[:SCALe]:OFFSet (on page 764)

Returns

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<NRf> The lower limit of the horizontal offset setting range.

Examples

`DISPLAY:IQVTIME:MEASview1:X:SCALE:OFFSET:MINIMUM?` might return `-16.28E-3`, indicating that the lower limit of the horizontal offset setting range is -16.28 ms.

DISPlay:IQVTime[:MEASview<y>]:Y[:SCALe]

Sets or queries the vertical range of the RF I&Q versus Time graph.

Conditions

Measurement views: RF I&Q versus Time

Group

Display commands

Syntax

```
DISPlay:IQVTime[:MEASview<y>]:Y[:SCALe] <value>
DISPlay:IQVTime[:MEASview<y>]:Y[:SCALe]?
```

Related Commands

DISPlay:IQVTime[:MEASview<y>]:Y[:SCALe]:OFFSet (on page 767)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<value>::<NRf> specifies the vertical range. Range: 1µ to 10 V.

Returns

<NRf>

Examples

DISPLAY:IQVTIME:MEASview1:Y:SCALE 1.5 sets the vertical range to 1.5 V in the RF I&Q versus Time graph.

DISPlay:IQVTime[:MEASview<y>]:Y[:SCALe]:AUTO (No Query Form)

Sets the vertical scale automatically to fit the waveform to the screen in the RF I&Q versus Time view.

Conditions

Measurement views: RF I&Q versus Time

Group

Display commands

Syntax

DISPlay:IQVTime[:MEASview<y>]:Y[:SCALe]:AUTO

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Examples

DISPLAY:IQVTIME:MEASview1:Y:SCALE:AUTO sets the vertical scale automatically to fit the waveform to the screen.

DISPlay:IQVTime[:MEASview<y>]:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (the value at the center of the vertical axis) in the RF I&Q versus Time graph.

Conditions

Measurement views: RF I&Q versus Time

Group

Display commands

Syntax

DISPlay:IQVTime[:MEASview<y>]:Y[:SCALe]:OFFSet <value>
DISPlay:IQVTime[:MEASview<y>]:Y[:SCALe]:OFFSet?

Related Commands

DISPlay:IQVTime[:MEASview<y>]:Y[:SCALe] (on page 766)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<value>::<NRf> specifies the vertical offset. Range: -5 to +5 V.

Returns

<NRf>

Examples

`DISPLAY:IQVTIME:MEASview1:Y:SCALE:OFFSET -82.75mV` sets the vertical offset to -82.75 mV in the RF I&Q versus Time graph.

DISPlay:IQVTime[:MEASview<y>]:Y[:SCALe]:RESCale (No Query Form)

Rescales the vertical axis automatically to fit the waveform to the screen.

Conditions

Measurement views: RF I&Q versus Time

Group

Display commands

Syntax

`DISPlay:IQVTime[:MEASview<y>]:Y[:SCALe]:RESCale`

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Examples

`DISPLAY:IQVTIME:MEASview1:Y:SCALE:RESCALE` rescales the vertical axis automatically to fit the waveform to the screen.

DISPlay:LTE:ACLR:MARKer:SHOW:STATe

Determines whether to show or hide the readout for the selected marker in the LTE ACLR display. It will enable or disable the checkbox in the Prefs tab of the LTE ACLR settings control panel. This command will not add the marker itself. It will only decide if the enabled marker readout is shown or hidden.

Conditions

Measurement view: LTE ACLR

Group

Display commands

Syntax

```
DISPlay:LTE:ACLR:MARKer:SHOW:STATe {OFF | ON | 1 | 0}
DISPlay:LTE:ACLR:MARKer:SHOW:STATe?
```

Arguments

`OFF` or 0 hides the readout for the selected marker in the LTE ACLR display.

`ON` or 1 shows the readout for the selected marker in the LTE ACLR display.

Returns

OFF or 0 means the readout for the selected marker in the LTE ACLR display is hidden.

ON or 1 means the readout for the selected marker in the LTE ACLR display is showing.

Examples

`DISPLAY:LTE:ACLR:MARKER:SHOW:STATE ON` shows the readout for the selected marker in the display if the marker is enabled.

DISPlay:LTE:ACLR:RESet:SCALe (No Query Form)

Resets the horizontal and vertical scale of the LTE ACLR display graph to the default values.

Conditions

Measurement view: LTE ACLR

Group

Display commands

Syntax

```
DISPlay:LTE:ACLR:RESet:SCALe
```

Arguments

None

Examples

`DISPLAY:LTE:ACLR:RESET:SCALE` resets the horizontal and vertical scale of the graph to the default values.

DISPlay:LTE:ACLR:SHOW:LIMit

Sets or queries the appearance of the limits in the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

Group

Display commands

Syntax

```
DISPlay:LTE:ACLR:SHOW:LIMit { SHADed | LINE | OFF }
DISPlay:LTE:ACLR:SHOW:LIMit?
```

Arguments

`SHADed`: denotes the limit area with shading.

`LINE`: delineates the limit area with a line.

`OFF`: turns off limits in the display.

Returns

SHADED: means that the limit area is denoted with shading.

LINE: means that the limit area is delineated with a line.

OFF: means that the display of limits is off.

Examples

`DISPLAY:LTE:ACLR:SHOW:LIMIT SHAD` sets the instrument to use shading to identify the limits of the graph in the display.

DISPlay:LTE:ACLR:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticules in the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

Group

Display commands

Syntax

```
DISPlay:LTE:ACLR:WINDow:TRACe:GRATicule:GRID:STATe {OFF | ON | 1 | 0}
DISPlay:LTE:ACLR:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or 0 hides the graticules in the graph.

`ON` or 1 shows the graticules in the graph.

Returns

0 indicates that the graticules in the graph is hidden.

1 indicates that the graticules in the graph is showing.

Examples

`DISPLAY:LTE:ACLR:WINDOW:TRACE:GRATICULE:GRID:STATE OFF` hides the graticules in the graph.

DISPlay:LTE:ACLR:WINDow:TRACe:POWer:LEVEl:STATe

Determines whether to show or hide the power readout in the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

Group

Display commands

Syntax

```
DISPlay:LTE:ACLR:WINDow:TRACe:POWer:LEVEl:STATe {OFF | ON | 1 | 0}
DISPlay:LTE:ACLR:WINDow:TRACe:POWer:LEVEl:STATe?
```

Arguments

`OFF` or 0 hides the power readout in the display.

`ON` or 1 shows the power readout in the display.

Returns

0 indicates that the power readout in the display is hidden.

1 indicates that the power readout in the display is showing.

Examples

`DISPLAY:LTE:ACLR:WINDOW:TRACE:POWER:LEVEL:STATE?` might return `1`, indicating that the power readout in the display is showing.

DISPlay:LTE:ACLR:X:SCALe

Sets or queries the horizontal range of the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

Group

Display commands

Syntax

```
DISPlay:LTE:ACLR:X:SCALe <value>
DISPlay:LTE:ACLR:X:SCALe?
```

Arguments

`<value>::<NRf>` specifies the horizontal range

Returns

`<NR3>`

Examples

`DISPLAY:LTE:ACLR:X:SCALE 10.000000000E+6` sets the horizontal range to 10 MHz.

DISPLay:LTE:ACLR:X:SCALe:AUTO (No Query Form)

Rescales the horizontal axis automatically to fit the waveform to the screen in the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

Group

Display commands

Syntax

```
DISPLay:LTE:ACLR:X:SCALe:AUTO
```

Arguments

None

Examples

`DISPLAY:LTE:ACLR:X:SCALE:AUTO` rescales the horizontal axis to fit the display screen.

DISPlay:LTE:ACLR:X:SCALe:OFFSet

Sets or queries the center frequency of the trace in the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

Group

Display commands

Syntax

```
DISPlay:LTE:ACLR:X:SCALe:OFFSet <value>
DISPlay:LTE:ACLR:X:SCALe:OFFSet?
```

Arguments

`<value>::=<NRf>` specifies the center frequency.

Returns

`<NR3>`

Examples

`DISPLAY:LTE:ACLR:X:SCAEL:OFFSET 1.450000000E+9` sets the center frequency to 1.45 GHz.

DISPlay:LTE:ACLR:Y:SCALe:AUTO (No Query Form)

Rescales the vertical scale automatically to fit the waveform on the screen in the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

Group

Display commands

Syntax

```
DISPlay:LTE:ACLR:Y:SCALe:AUTO
```

Arguments

None

Examples

`DISPLAY:LTE:ACLR:Y:SCALE:AUTO` rescales the vertical scale to fit the waveform on the screen.

DISPlay:LTE:ACLR:Y:SCALe:OFFSet

Sets or queries the vertical offset of the trace in the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

Group

Display commands

Syntax

```
DISPlay:LTE:ACLR:Y:SCALe:OFFSet <value>
```

```
DISPlay:LTE:ACLR:Y:SCALe:OFFSet?
```

Arguments

`<value>::=<NRf>` specifies the vertical offset.

Returns

`<NRf>`

Examples

`DISPLAY:LTE:ACLR:Y:SCALE:OFFSET 40` sets the vertical position of the trace to 40 dBm.

DISPlay:LTE:ACLR:Y:SCALe

Sets or queries the vertical range of the graph in the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

Group

Display commands

Syntax

```
DISPlay:LTE:ACLR:Y:SCALe <value>
DISPlay:LTE:ACLR:Y:SCALe?
```

Arguments

`<value>::=<NRf>` specifies the vertical range.

Returns

`<NRf>`

Examples

`DISPLAY:LTE:ACLR:Y:SCALE 100` sets the vertical range to 100 dB.

DISPlay:LTE:CHSPectrum:MARKer:SHOW:STATe

Determines whether to show or hide the readout for the selected marker in the LTE Channel Spectrum display. It will enable or disable the checkbox in the Prefs tab of the LTE Channel Spectrum settings control panel. This command will not add the marker itself. It will only decide if the enabled marker readout is shown or hidden.

Conditions

Measurement view: LTE Channel Spectrum

Group

Display commands

Syntax

```
DISPlay:LTE:CHSPectrum:MARKer:SHOW:STATe {OFF | ON | 1 | 0}
DISPlay:LTE:CHSPectrum:MARKer:SHOW:STATe?
```

Arguments

OFF or 0 hides the readout for the selected marker in the LTE Channel Spectrum display.

ON or 1 shows the readout for the selected marker in the LTE Channel Spectrum display.

Returns

OFF or 0 means the readout for the selected marker in the LTE Channel Spectrum display is hidden.

ON or 1 means the readout for the selected marker in the LTE Channel Spectrum display is showing.

Examples

DISPLAY:LTE:CHSPECTRUM:MARKER:SHOW:STATE ON shows the readout for the selected marker in the display if the marker is enabled.

DISPlay:LTE:CHSPectrum:RESet:SCALe (No Query Form)

Resets the horizontal and vertical scale of the LTE Channel Spectrum display graph to the default values.

Conditions

Measurement view: LTE Channel Spectrum

Group

Display commands

Syntax

```
DISPlay:LTE:CHSPectrum:RESet:SCALe
```

Arguments

None

Examples

DISPLAY:LTE:CHSPECTRUM:RESET:SCALE resets the horizontal and vertical scale of the graph to the default values.

DISPlay:LTE:CHSPectrum:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule in the LTE Channel Spectrum display.

Conditions

Measurement view: LTE Channel Spectrum

Group

Display commands

Syntax

```
DISPlay:LTE:CHSPectrum:WINDow:TRACe:GRATicule:GRID:STATe {OFF | ON | 1 | 0}
DISPlay:LTE:CHSPectrum:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

OFF or 0 hides the graticule in the LTE Channel Spectrum display.

ON or 1 shows the graticule in the LTE Channel Spectrum display.

Returns

OFF or 0 means the graticule in the LTE Channel Spectrum display is hidden.

ON or 1 means the graticule in the LTE Channel Spectrum display is showing.

Examples

`DISPLAY:LTE:CHSPECTRUM:WINDOW:TRACE:GRATICULE:GRID:STATE OFF` hides the graticules in the LTE Channel Spectrum display.

DISPlay:LTE:CHSPectrum:X:SCALe

Sets or queries the horizontal range of the LTE Channel Spectrum display.

Conditions

Measurement view: LTE Channel Spectrum

Group

Display commands

Syntax

```
DISPlay:LTE:CHSPectrum:X:SCALe <value>
DISPlay:LTE:CHSPectrum:X:SCALe?
```

Arguments

`<value>::<NRf>` specifies the horizontal range

Returns

`<NR3>`

Examples

`DISPLAY:LTE:CHSPECTRUM:X:SCALE 10.000000000E+6` sets the horizontal range to 10 MHz.

DISPLay:LTE:CHSpectrum:X:SCALe:AUTO (No Query Form)

Rescales the horizontal axis automatically to fit the waveform to the screen in the LTE Channel Spectrum display.

Conditions

Measurement view: LTE Channel Spectrum

Group

Display commands

Syntax

```
DISPLay:LTE:CHSpectrum:X:SCALe:AUTO
```

Arguments

None

Examples

`DISPLAY:LTE:CHSPECTRUM:X:SCALE:AUTO` rescales the horizontal axis to fit the display screen.

DISPlay:LTE:CHSPectrum:X:SCALe:OFFSet

Sets or queries the center frequency of the trace in the LTE Channel Spectrum display.

Conditions

Measurement view: LTE Channel Spectrum

Group

Display commands

Syntax

```
DISPlay:LTE:CHSPectrum:X:SCALe:OFFSet <value>
DISPlay:LTE:CHSPectrum:X:SCALe:OFFSet?
```

Arguments

<value>::=<NRf> specifies the center frequency.

Returns

<NR3>

Examples

`DISPLAY:LTE:ACLR:X:SCAEL:OFFSET 1.450000000E+9` sets the center frequency to 1.45 GHz.

DISPlay:LTE:CHSPectrum:Y:SCALe

Sets or queries the vertical range of the graph in the LTE Channel Spectrum display.

Conditions

Measurement view: LTE Channel Spectrum

Group

Display commands

Syntax

```
DISPlay:LTE:CHSPectrum:Y:SCALe <value>
DISPlay:LTE:CHSPectrum:Y:SCALe?
```

Arguments

`<value>::=<NRf>` specifies the vertical range.

Returns

`<NRf>`

Examples

`DISPLAY:LTE:CHSPECTRUM:Y:SCALE 100` sets the vertical range to 100 dB.

DISPlay:LTE:CHSPectrum:Y:SCALe:AUTO (No Query Form)

Rescales the vertical scale automatically to fit the waveform on the screen in the LTE Channel Spectrum display.

Conditions

Measurement view: LTE Channel Spectrum

Group

Display commands

Syntax

```
DISPlay:LTE:CHSPectrum:Y:SCALe:AUTO
DISPlay:LTE:CHSPectrum:Y:SCALe:AUTO?
```

Arguments

None

Examples

`DISPLAY:LTE:CHSPECTRUM:Y:SCALE:AUTO` rescales the vertical scale to fit the waveform on the screen.

DISPlay:LTE:CHSPectrum:Y:SCALe:OFFSet

Sets or queries the vertical offset of the trace in the LTE Channel Spectrum display.

Conditions

Measurement view: LTE Channel Spectrum

Group

Display commands

Syntax

```
DISPlay:LTE:CHSPectrum:Y:SCALe:OFFSet <value>
DISPlay:LTE:CHSPectrum:Y:SCALe:OFFSet?
```

Arguments

`<value>::=<NRf>` specifies the vertical offset.

Returns

`<NRf>`

Examples

`DISPLAY:LTE:CHSPECTRUM:Y:SCALE:OFFSET 40` sets the vertical position of the trace to 40 dBm.

DISPlay:LTE:CONSte:SCALe

Sets or queries the scale (zoom) in the LTE Constellation display.

Conditions

Measurement view: LTE Constellation

Group

Display commands

Syntax

```
DISPlay:LTE:CONSte:SCALe <value>
DISPlay:LTE:CONSte:SCALe?
```

Arguments

<value>::<NRf> specifies the scale (zoom)

Returns

<NRf>

Examples

DISPLAY:LTE:CONSTE:SCALE 5 zooms the display out by 5.

DISPlay:LTE:CONSte:X:OFFSet

Sets or queries the horizontal offset in the LTE Constellation display.

Conditions

Measurement view: LTE Constellation

Group

Display commands

Syntax

```
DISPlay:LTE:CONSte:X:OFFSet <value>
DISPlay:LTE:CONSte:X:OFFSet?
```

Arguments

<value>::<NRf> specifies the horizontal offset range.

Returns

<NRf>

Examples

DISPLAY:LTE:CONSTE:X:OFFSET 2.7 sets the horizontal offset to 2.7 units to the left.

DISPlay:LTE:CONSte:Y:OFFSet

Sets or queries the vertical offset in the LTE Constellation display.

Conditions

Measurement view: LTE Constellation

Group

Display commands

Syntax

```
DISPlay:LTE:CONSte:Y:OFFSet <value>
DISPlay:LTE:CONSte:Y:OFFSet?
```

Arguments

<value>::<NRf> specifies the vertical offset range. A positive value moves the offset up. A negative value moves the offset down and the display moves in the other direction.

Returns

<NRf>

Examples

DISPLAY:LTE:CONSTE:Y:OFFSET 2 sets the vertical offset to 2 units.

DISPlay:LTE:MEASview:DELete (No Query Form)

Closes the selected LTE measurement display in the application.

Conditions

Measurement view: Any LTE display

Group

Display commands

Syntax

DISPlay:LTE:MEASview:DELete { CONS | ACLR | CHSP | PVT }

Arguments

CONS closes the LTE Constellation display.

ACLR closes the LTE ACLR display.

CHSP closes the LTE Channel Spectrum display.

PVT closes the LTE Power vs Time display.

Examples

DISPLAY:LTE:MEASVIEW:DELETE ACLR closes the LTE ACLR display.

DISPlay:LTE:MEASview:NEW (No Query Form)

Opens a new LTE measurement display.

Conditions

Measurement view: Any LTE display

Group

Display commands

Syntax

DISPlay:LTE:MEASview:NEW { CONS | ACLR | CHSP | PVT }

Arguments

CONS opens the LTE Constellation display.

ACLR opens the LTE ACLR display.

CHSP opens the LTE Channel Spectrum display.

PVT opens the LTE Power vs Time display.

Examples

DISPLAY:LTE:MEASVIEW:NEW ACLR opens a new LTE ACLR display.

DISPlay:LTE:MEASview:SELect

Selects an LTE measurement display in the application. The query returns the currently selected display.

Conditions

Measurement view: Any LTE display

Group

Display commands

Syntax

```
DISPlay:LTE:MEASview:SELect { CONS | ACLR | CHSP | PVT }
DISPlay:LTE:MEASview:SELect?
```

Arguments

CONS selects the LTE Constellation display.

ACLR selects the LTE ACLR display.

CHSP selects the LTE Channel Spectrum display.

PVT selects the LTE Power vs Time display.

Returns

CONS means the LTE Constellation display is selected.

ACLR means the LTE ACLR display is selected.

CHSP means the LTE Channel Spectrum display is selected.

PVT means the LTE Power vs Time display is selected.

Examples

DISPLAY:LTE:MEASVIEW:SELECT ACLR selects the LTE ACLR display.

DISPlay:LTE:PVTime:MARKer:SHOW:STATe

Shows or hides the marker readout on the LTE Power vs. Time display.

Conditions

Measurement view: LTE Power vs Time

Group

Display commands

Syntax

```
DISPlay:LTE:PVTime:MARKer:SHOW:STATe {OFF | ON | 0 | 1 }
DISPlay:LTE:PVTime:MARKer:SHOW:STATe?
```

Arguments

OFF or 0 hides the marker readout in the LTE Power vs Time display.

ON or 1 shows the marker readout in the LTE Power vs Time display.

Returns

OFF or 0 means the marker readout in the LTE Power vs Time display is hidden.

ON or 1 means the marker readout in the LTE Power vs Time display is showing.

Examples

DISPLAY:LTE:PVTIME:MARKER:SHOW:STATE ON turns on the marker readout.

DISPlay:LTE:PVTime:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule in the LTE Power vs Time display.

Conditions

Measurement view: LTE Power vs Time

Group

Display commands

Syntax

```
DISPlay:LTE:PVTime:WINDow:TRACe:GRATicule:GRID:STATe {OFF | ON | 0 | 1 }
DISPlay:LTE:PVTime:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

OFF or 0 hides the graticule in the LTE Power vs Time display.

ON or 1 shows the graticule in the LTE Power vs Time display.

Returns

OFF or 0 means the graticule in the LTE Power vs Time display is hidden.

ON or 1 means the graticule in the LTE Power vs Time display is showing.

Examples

DISPLAY:LTE:PVTIME:WINDOW:TRACE:GRATICULE:GRID:STATE OFF hides the graticules in the LTE Power vs Time display.

DISPlay:LTE:PVTime:X:SCALe

Sets or queries the vertical range of the LTE Power vs Time graph.

Conditions

Measurement view: LTE Power vs Time

Group

Display commands

Syntax

```
DISPlay:LTE:PVTime:X:SCALe <NRf>
DISPlay:LTE:PVTime:X:SCALe?
```

Arguments

<NRf> specifies the vertical range.

Returns

<NRf>

Examples

DISPLAY:LTE:PVTIME:X:SCALE 10 sets the horizontal range to 10 dB.

DISPLay:LTE:PVTime:X:SCALe:AUTO (No Query Form)

Rescales the horizontal axis automatically to fit the waveform to the screen in the LTE Power vs Time display.

Conditions

Measurement view: LTE Power vs Time

Group

Display commands

Syntax

DISPLay:LTE:PVTime:X:SCALe:AUTO

Arguments

None

Examples

DISPLAY:LTE:PVTIME:X:SCALE:AUTO rescales the horizontal axis to fit the display screen.

DISPlay:LTE:PVTime:X:SCALe:RESet (No Query Form)

Resets the horizontal scale to the default value for the time measurements for the LTE Power vs Time display.

Conditions

Measurement view: LTE Power vs Time

Group

Display commands

Syntax

DISPlay:LTE:PVTime:X:SCALe:RESet

Arguments

None

Examples

DISPLAY:LTE:PVTIME:X:SCALE:RESET resets the horizontal scale to the default value.

DISPlay:LTE:PVTime:Y:SCALe

Sets or queries the horizontal range of the LTE Power vs Time graph.

Conditions

Measurement view: LTE Power vs Time

Group

Display commands

Syntax

```
DISPlay:LTE:PVTime:Y:SCALe <value>
DISPlay:LTE:PVTime:Y:SCALe?
```

Arguments

<value>::<NRf> specifies the horizontal range.

Returns

<NRf>

Examples

DISPLAY:LTE:PVTIME:Y:SCALE 100 sets the horizontal range to 100 ms.

DISPlay:LTE:PVTime:Y:SCALe:AUTO (No Query Form)

Rescales the vertical scale automatically to fit the waveform on the screen in the LTE Power vs Time display.

Conditions

Measurement view: LTE Power vs Time

Group

Display commands

Syntax

```
DISPlay:LTE:PVTime:Y:SCALe:AUTO
```

Arguments

None

Examples

DISPLAY:LTE:PVTIME:Y:SCALE:AUTO rescales the vertical scale to fit the waveform on the screen.

DISPlay:LTE:PVTime:Y:SCALe:OFFSet

Sets or queries the vertical offset (top edge of the vertical axis) of the trace in the LTE Power vs Time display.

Conditions

Measurement view: LTE Power vs Time

Group

Display commands

Syntax

```
DISPlay:LTE:PVTime:Y:SCALe:OFFSet <NRf>
DISPlay:LTE:PVTime:Y:SCALe:OFFSet?
```

Arguments

<NRf> specifies the vertical offset.

Returns

<NRf>

Examples

DISPLAY:LTE:PVTIME:Y:SCALE:OFFSET −10 sets the vertical position of the trace to −10 dBm.

DISPlay:LTE:PVTime:Y:SCALe:PDIVision

Sets or queries the value of the vertical scale for the LTE Power vs. Time display in dB/division. This is only a visual control for panning the graph.

Conditions

Measurement view: LTE Power vs Time

Group

Display commands

Syntax

```
DISPlay:LTE:PVTime:Y:SCALe:PDIVision <NRf>
DISPlay:LTE:PVTime:Y:SCALe:PDIVision?
```

Arguments

<NRf> specifies the vertical scale in dB/div.

Returns

<NRf>

Examples

DISPLAY:LTE:PVTIME:Y:SCALE:PDIVISION 20 sets the vertical scale to 20.0 dB/div.

DISPlay:LTE:PVTime:Y:SCALe:RESet (No Query Form)

Resets the vertical scale and position (offset) to the default value for the time measurements for the LTE Power vs Time display.

Conditions

Measurement view: LTE Power vs Time

Group

Display commands

Syntax

```
DISPlay:LTE:PVTime:Y:SCALe:RESet
```

Arguments

None

Examples

DISPLAY:LTE:PVTIME:Y:SCALE:RESET resets the vertical scale and position (offset) to the default value.

DISPlay:MCPower:MARKer:SHOW:STATe

Determines whether to show or hide the readout for the selected marker in the MCPR view.

Conditions

Measurement views: MCPR

Group

Display commands

Syntax

```
DISPlay:MCPower:MARKer:SHOW:STATe { OFF | ON | 0 | 1 }
DISPlay:MCPower:MARKer:SHOW:STATe?
```

Arguments

OFF or 0 hides the readout for the selected marker in the graph.

ON or 1 shows the readout for the selected marker in the graph.

Examples

DISPLAY:MCPOWER:MARKER:SHOW:STATE ON shows the readout for the selected marker in the graph.

DISPlay:MCPower:PLEVel:SHOW:STATe

Determines whether to show or hide the power levels in the MCPR view.

Conditions

Measurement views: MCPR

Group

Display commands

Syntax

```
DISPlay:MCPower:PLEVel:SHOW:STATe { OFF | ON | 0 | 1 }
DISPlay:MCPower:PLEVel:SHOW:STATe?
```

Arguments

OFF or 0 hides the power levels in the graph.

ON or 1 shows the power levels in the graph.

Examples

DISPLAY:MCPOWER:PLEVEL:SHOW:STATE ON shows the power levels in the graph.

DISPlay:MCPower:RESet:SCALe (No Query Form)

Resets the horizontal and vertical scale to the default values described below in the MCPR view.

Vertical offset = Reference level, Vertical scale = 100 dB,

Horizontal offset = Center frequency, and Horizontal scale = Default span

Conditions

Measurement views: MCPR

Group

Display commands

Syntax

```
DISPlay:MCPower:RESet:SCALe
```

Arguments

None

Examples

`DISPLAY:MCPOWER:RESET:SCALE` resets the horizontal and vertical scale to the default values.

DISPlay:MCPower:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions

Measurement views: MCPR

Group

Display commands

Syntax

```
DISPlay:MCPower:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:MCPower:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or 0 hides the graticule grid.

`ON` or 1 shows the graticule grid.

Examples

`DISPLAY:MCPOWER:WINDOW:TRACE:GRATICULE:GRID:STATE ON` shows the graticule grid on the screen.

DISPlay:MCPower:X[:SCALe]

Sets or queries the horizontal range of the MCPR graph.

Conditions

Measurement views: MCPR

Group

Display commands

Syntax

```
DISPlay:MCPower:X[:SCALe] <value>
DISPlay:MCPower:X[:SCALe]?
```

Related Commands

DISPlay:MCPower:X[:SCALe]:OFFSet (on page 787)

Arguments

<value>::<NRf> specifies the horizontal range.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples

DISPLAY:MCPOWER:X:SCALE 10MHz sets the horizontal range to 10 MHz.

DISPlay:MCPower:X[:SCALe]:AUTO (No Query Form)

Rescales the horizontal axis automatically to fit the waveform to the screen in the MCPR view.

Conditions

Measurement views: MCPR

Group

Display commands

Syntax

DISPlay:MCPower:X[:SCALe]:AUTO

Arguments

None

Examples

DISPLAY:MCPOWER:X:SCALE:AUTO rescales the horizontal scale automatically to fit the waveform to the screen.

DISPlay:MCPower:X[:SCALe]:OFFSet

Sets or queries the minimum horizontal value (left edge) of the MCPR graph.

Conditions

Measurement views: MCPR

Group

Display commands

Syntax

DISPlay:MCPower:X[:SCALe]:OFFSet <value>
DISPlay:MCPower:X[:SCALe]:OFFSet?

Related Commands

DISPlay:MCPower:X[:SCALe] (on page 786)

Arguments

<value>::<NRf> specifies the minimum horizontal value.

Range: [(center frequency) - (X scale) × 0.9] to [(center frequency) + (X scale) × 0.9]

Examples

DISPLAY:MCPOWER:X:SCALE:OFFSET 1.45GHz sets the minimum horizontal value to 1.45 GHz in the MCPR graph.

DISPlay:MCPower:Y[:SCALe]

Sets or queries the vertical range of the MCPR graph.

Conditions

Measurement views: MCPR

Group

Display commands

Syntax

```
DISPlay:MCPower:Y[:SCALe] <value>
DISPlay:MCPower:Y[:SCALe]?
```

Related Commands

DISPlay:MCPower:Y[:SCALe]:OFFSet (on page 789)

Arguments

<value>::<NRf> specifies the vertical range. Range: 0.1 to 200 dB.

Examples

DISPLAY:MCPOWER:Y:SCALE 100 sets the vertical range to 100 dB in the MCPR graph.

DISPlay:MCPower:Y[:SCALe]:AUTO (No Query Form)

Rescales the vertical axis automatically to fit the waveform to the screen in the MCPR view.

Conditions

Measurement views: MCPR

Group

Display commands

Syntax

```
DISPlay:MCPower:Y[:SCALe]:AUTO
```

Arguments

None

Examples

DISPLAY:MCPOWER:Y:SCALE:AUTO rescales the vertical scale automatically to fit the waveform to the screen.

DISPlay:MCPower:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (the value at the top edge of the vertical axis) in the MCPR graph.

Conditions

Measurement views: MCPR

Group

Display commands

Syntax

```
DISPlay:MCPower:Y[:SCALe]:OFFSet <value>
DISPlay:MCPower:Y[:SCALe]:OFFSet?
```

Related Commands

DISPlay:MCPower:Y[:SCALe] (on page 788)

Arguments

`<value>::<NRf>` specifies the vertical offset. Range: -170 to +50 dBm.

Examples

`DISPLAY:MCPOWER:Y:SCALE:OFFSET -12.5` sets the vertical offset to -12.5 dBm in the MCPR graph.

DISPlay:MERRor[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe

Sets or queries the graticule grid view state for the Magnitude error versus Time view.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Magnitude error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Magnitude error versus Time).

Group

Display commands

Syntax

```
DISPlay:MERRor[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:MERRor[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or 0 hides the graticule grid.

`ON` or 1 shows the graticule grid.

Returns

OFF or 0

ON or 1

Examples

`DISPLay:MERRor:MEASview2:WINDow:TRACe:GRATicule:GRID:STATe ON` shows the graticule grid on the Magnitude error versus Time Measview 2.

DISPlay:MERRor[:MEASview<y>]:Y[:SCALe]

Sets or queries the vertical range of the Magnitude error versus Time graph.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Magnitude error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Magnitude error versus Time).

Group

Display commands

Syntax

```
DISPlay:MERRor[:MEASview<y>]:Y[:SCALe] <value>
DISPlay:MERRor[:MEASview<y>]:Y[:SCALe]?
```

Related Commands

DISPlay:MERRor:[MEASview<y>]:Y[:SCALe]:OFFSet (on page 791)

Arguments

`<value>::<NRf>` specifies the vertical range.

Range: 1 to 100%.

Returns

`<NRF>`

Examples

`DISPlay:MERRor:MEASview1:Y:SCALe 50` sets the vertical range to 50% in the Magnitude error versus Time display Measurement view 1.

DISPlay:MERRor[:MEASview<y>]:Y[:SCALe]:AUTO (No query form)

Sets the vertical scale automatically to fit the waveform to the screen in the Magnitude error versus Time display.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Magnitude error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Magnitude error versus Time).

Group

Display commands

Syntax

```
DISPlay:MERRor[:MEASview<y>]:Y[:SCALe]:AUTO
```

Arguments

None

Examples

`DISPLay:MERRor:MEASview2:Y:SCALe:AUTO` sets the vertical scale automatically to fit the waveform to the screen of Magnitude error versus Time Measurement view 2.

DISPlay:MERRor[:MEASview<y>]:Y[:SCALe]:OFFSet

Sets or queries the minimum vertical value (bottom edge) of the Magnitude error versus Time graph.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Magnitude error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Magnitude error versus Time).

Group

Display commands

Syntax

```
DISPlay:MERRor[:MEASview<y>]:Y[:SCALe]:OFFSet <value>
DISPlay:MERRor[:MEASview<y>]:Y[:SCALe]:OFFSet?
```

Related Commands

DISPlay:MERRor[:MEASview<y>]:Y[:SCALe] (on page 790)

Arguments

`<value>::<NRf>` specifies the minimum vertical value.

Range: -100 to +100%.

Returns

`<NRF>`

Examples

`DISPlay:MERRor:MEASview2:Y:SCALe:OFFSet? -9.5` sets the minimum vertical value to -9.5% in the Magnitude error versus Time graph Measurement View 2.

DISPlay:NOISe:FIGure[:SCALe]:AUTO (No Query Form)

Automatically scales the Noise Figure graph.

Conditions

Measurement view: Noise Figure

Group

Display commands

Syntax

```
DISPlay:NOISe:FIGure[:SCALe]:AUTO
```

Arguments

None

Examples

`DISP:NOIS:FIG:SCAL:AUTO` will automatically scale the Noise Figure graph.

DISPlay:NOISe:FIGure:WINDow:TRACe:GRATicule:GRID:STATe

Sets or queries the graticule state in the Noise Figure display to on (showing) or off (hidden).

Conditions

Measurement view: Noise Figure

Group

Display commands

Syntax

```
DISPlay:NOISe:FIGure:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:NOISe:FIGure:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or 0 hides the graticule in the Noise Figure display.

`ON` or 1 shows the graticule in the Noise Figure display.

Returns

0 means the graticule in the Noise Figure display is hidden (off).

1 means the graticule in the Noise Figure display is showing (on).

Examples

`DISPLAY:NOISE:FIGURE:WINDOW:TRACE:GRATICULE:GRID:STATE ON` will turn on the graticule in the Noise Figure display.

`DISPLAY:NOISE:FIGURE:WINDOW:TRACE:GRATICULE:GRID:STATE?` return of 1 indicates that the graticule in the Noise Figure display is showing (on).

DISPlay:NOISe:FIGure:WINDow:TRACe:LEGend:STATe

Sets or queries the trace legend state in the Noise Figure display to on (showing) or off (hidden).

Conditions

Measurement view: Noise Figure

Group

Display commands

Syntax

```
DISPlay:NOISe:FIGure:WINDow:TRACe:LEGend:STATe { OFF | ON | 0 | 1 }
DISPlay:NOISe:FIGure:WINDow:TRACe:LEGend:STATe?
```

Arguments

OFF or 0 hides the trace legend in the Noise Figure display.

ON or 1 shows the trace legend in the Noise Figure display.

Returns

0 means the trace legend in the Noise Figure display is hidden (off).

1 means the trace legend in the Noise Figure display is showing (on).

Examples

DISPLAY:NOISE:FIGURE:WINDOW:TRACE:LEGEND:STATE ON will turn on the trace legend in the Noise Figure display.

DISPLAY:NOISE:FIGURE:WINDOW:TRACE:LEGEND:STATE? return of 1 indicates that the trace legend in the Noise Figure display is showing (on).

DISPlay:NOISe:FIGure:WINDow:TRACe:MEASpoints:STATe

Sets or queries the measurement points state in the Noise Figure display to on (showing) or off (hidden).

Conditions

Measurement view: Noise Figure

Group

Display commands

Syntax

```
DISPlay:NOISe:FIGure:WINDow:TRACe:MEASpoints:STATe { OFF | ON | 0 | 1 }
DISPlay:NOISe:FIGure:WINDow:TRACe:MEASpoints:STATe?
```

Arguments

OFF or 0 hides the measurement points in the Noise Figure display.

ON or 1 shows the measurement points in the Noise Figure display.

Returns

0 means the measurement points in the Noise Figure display is hidden (off).

1 means the measurement points in the Noise Figure display is showing (on).

Examples

DISPLAY:NOISE:FIGURE:WINDOW:TRACE:MEASPOINTS:STATE ON will turn on the measurement points in the Noise Figure display.

DISPLAY:NOISE:FIGURE:WINDOW:TRACE:MEASPOINTS:STATE? might return 1, indicating that the measurement points in the Noise Figure display are showing (on).

DISPlay:NOISe:FIGure:X:AUTO (No Query Form)

Automatically sets the horizontal scale of the Noise Figure graph.

Conditions

Measurement view: Noise Figure

Group

Display commands

Syntax

```
DISPlay:NOISe:FIGure:X:AUTO
```

Arguments

None

DISPlay:NOISe:FIGure:X:OFFSet

Sets or queries the center frequency (offset) in the Noise Figure display.

Conditions

Measurement view: Noise Figure

Group

Display commands

Syntax

```
DISPlay:NOISe:FIGure:X:OFFSet <NRf>
DISPlay:NOISe:FIGure:X:OFFSet?
```

Arguments

<NRf> is the center frequency (offset) in the Noise Figure display.

Range: -1.9900 GHz to +5.0000 GHz

Returns

<NR3> is a floating point value with an exponent. This is the center frequency (offset) value.

Examples

DISPLAY:NOISE:FIGURE:X:OFFSET? might return 2.0000000009E+9, indicating that the center frequency (offset) value is 2.0000 GHz.

DISPlay:NOISe:FIGure:X[:SCALe]

Sets or queries the horizontal scale of the Noise Figure display.

Conditions

Measurement view: Noise Figure

Group

Display commands

Syntax

```
DISPlay:NOISe:FIGure:X[:SCALe] <NRf>
DISPlay:NOISe:FIGure:X[:SCALe]
```

Arguments

<NRf> is the horizontal scale value of the display.

Range: 100 Hz to 6.2 GHz

Returns

<NR3> = a floating point value with an exponent. This is the horizontal scale value.

Examples

DISPLAY:NOISE:FIGURE:X:SCALE? might return 2.9900000000E+9, indicating that the horizontal scale value is 3.0 GHz.

DISPlay:NOISe:FIGure:Y:AUTO (No Query Form)

Automatically sets the vertical scale of the Noise Figure graph.

Conditions

Measurement view: Noise Figure

Group

Display commands

Syntax

```
DISPlay:NOISe:FIGure:Y:AUTO
```

Arguments

None

DISPlay:NOISe:FIGure:Y:AUTO:STATe

Sets the state of the automatic vertical scale and position of the graph to on or off. The query form of the command returns the current state.

Conditions

Measurement view: Noise Figure

Group

Display commands

Syntax

```
DISPlay:NOISe:FIGure:Y:AUTO:STATe { OFF | ON | 0 | 1 }
DISPlay:NOISe:FIGure:Y:AUTO:STATe?
```

Arguments

OFF or 0 sets the analyzer to manual setting of the vertical scale and position of the graph.

ON or 1 sets the analyzer to automatically set the vertical scale and position of the graph.

Returns

OFF or 0 means the analyzer is in manual mode for setting of the vertical scale and position of the graph.

ON or 1 means the analyzer is in automatic mode for setting of the vertical scale and position of the graph.

Examples

`DISPLAY:NOISE:FIGURE:Y:AUTO:STATE ON` sets the analyzer to automatically set the vertical scale and position of the graph.

DISPlay:NOISe:FIGure:Y:PDIVision

Sets or queries the distance between graticule lines on the display in dB.

Conditions

Measurement view: Noise Figure

Group

Display commands

Syntax

```
DISPlay:NOISe:FIGure:Y:PDIVision <NRf>
DISPlay:NOISe:FIGure:Y:PDIVision?
```

Arguments

`<NRf>` is the distance between graticule lines on the display.

Range: 100 mdB to 5.00 dB

Returns

`<NR2>` = a floating point value without an exponent, which is the distance between graticule lines on the display in dB.

Examples

`DISPLAY:NOISE:FIGURE:Y:PDIVISION?` might return 1.5000000000, indicating that the distance between graticule lines on the display is 1.50 dB.

DISPlay:NOISe:FIGure:Y:POSition

Sets or queries the vertical position of the Noise Figure display.

Conditions

Measurement view: Noise Figure

Group

Display commands

Syntax

```
DISPlay:NOISe:FIGure:Y:POSition <NRf>
DISPlay:NOISe:FIGure:Y:POSition?
```

Arguments

`<NRf>` is the vertical position of the Noise Figure display.

Range: 5.12 to 26.5

Returns

`<NR2>` =a floating point value without an exponent, which is the vertical position value.

The vertical position value

Examples

`DISPLAY:NOISE:FIGURE:Y:POSITION?` might return 26.5000000000, indicating that the vertical position is 26.5.

DISPlay:NOISe:FIGure:Y:SCALe

Sets or queries the vertical scale of the Noise Figure display.

Conditions

Measurement view: Noise Figure

Group

Display commands

Syntax

```
DISPlay:NOISe:FIGure:Y:SCALe <NRf>
DISPlay:NOISe:FIGure:Y:SCALe?
```

Arguments

`<NRf>` is the vertical scale of the Noise Figure display Range: 10.2 to 31.6

Returns

`<NR2>` = a floating point value without an exponent, which is the vertical scale value.

Examples

`DISPLAY:NOISE:FIGURE:Y:SCALE?` might return 10.0000000000, indicating that the vertical scale is 10.0 dB.

DISPlay:NOISe:FIGure:Y:SCALe:LINear:STATe

Sets or queries whether the linear units are shown (on) or not shown (off) on the display.

Conditions

Measurement view: Noise Figure

Group

Display commands

Syntax

```
DISPlay:NOISe:FIGure:Y:SCALe:LINear:STATe { OFF | ON | 0 | 1 }
DISPlay:NOISe:FIGure:Y:SCALe:LINear:STATe?
```

Arguments

`OFF` or 0 sets the analyzer to show the linear units on the display.

`ON` or 1 sets the analyzer to not show the linear units on the display.

Returns

OFF or 0 indicates that the analyzer is set to show the linear units on the display.

ON or 1 indicates that the analyzer is set to not show the linear units on the display.

Examples

DISPLAY:NOISE:FIGURE:Y:SCALE:LINEAR:STATE? might return 1, indicating that linear units are not showing on the display.

DISPlay:NOISe:GAIN[:SCALe]:AUTO (No Query Form)

Automatically scales the Gain graph.

Conditions

Measurement view: Gain

Group

Display commands

Syntax

DISPlay:NOISe:GAIN[:SCALe]:AUTO

Arguments

None

DISPlay:NOISe:GAIN:WINDow:TRACe:GRATicule:GRID:STATe

Sets or queries the graticule state in the Gain display to on (showing) or off (hidden).

Conditions

Measurement view: Gain

Group

Display commands

Syntax

DISPlay:NOISe:GAIN:WINDow:TRACe:GRATicule:GRID:STATe {OFF|ON|0|1}
DISPlay:NOISe:GAIN:WINDow:TRACe:GRATicule:GRID:STATe?

Arguments

OFF or 0 hides the graticule in the Gain display.

ON or 1 shows the graticule in the Gain display.

Returns

0 means the graticule in the Gain display is hidden (off).

1 means the graticule in the Gain display is showing (on).

Examples

DISPLAY:NOISE:GAIN:WINDOW:TRACE:GRATICULE:GRID:STATE ON will turn on the graticule in the Gain display.

DISPLAY:NOISE:GAIN:WINDOW:TRACE:GRATICULE:GRID:STATE? might return 1, indicating that the graticule in the Gain display is showing (on).

DISPlay:NOISe:GAIN:WINDow:TRACe:LEGend:STATe

Sets or queries the trace legend state in the Gain display to on (showing) or off (hidden).

Conditions

Measurement view: Gain

Group

Display commands

Syntax

```
DISPlay:NOISe:GAIN:WINDow:TRACe:LEGend:STATe { OFF | ON | 0 | 1 }
DISPlay:NOISe:GAIN:WINDow:TRACe:LEGend:STATe?
```

Arguments

`OFF` or 0 hides the trace legend in the Gain display.

`ON` or 1 shows the trace legend in the Gain display.

Returns

0 means the trace legend in the Gain display is hidden (off).

1 means the trace legend in the Gain display is showing (on).

Examples

`DISPLAY:NOISE:GAIN:WINDOW:TRACE:LEGEND:STATE ON` will turn on the trace legend in the Gain display.

`DISPLAY:NOISE:GAIN:WINDOW:TRACE:LEGEND:STATE? 1` indicates that the trace legend in the Gain display is showing (on).

DISPlay:NOISe:GAIN:WINDow:TRACe:MEASPoints:STATe

Sets or queries the measurement points state in the Gain display to on (showing) or off (hidden).

Conditions

Measurement view: Gain

Group

Display commands

Syntax

```
DISPlay:NOISe:GAIN:WINDow:TRACe:MEASPoints:STATe {OFF|ON|0|1}
DISPlay:NOISe:GAIN:WINDow:TRACe:MEASPoints:STATe?
```

Arguments

`OFF` or 0 hides the measurement points in the Gain display.

`ON` or 1 shows the measurement points in the Gain display.

Returns

0 means the measurement points in the Gain display is hidden (off).

1 means the measurement points in the Gain display is showing (on).

Examples

`DISPLAY:NOISE:GAIN:WINDOW:TRACE:MEASPOINTS:STATE ON` will turn on the measurement points in the Gain display.

`DISPLAY:NOISE:GAIN:WINDOW:TRACE:MEASPOINTS:STATE?` might return `1`, indicating that the measurement points in the Gain display are showing (on).

DISPlay:NOISe:GAIN:X:AUTO (No Query Form)

Automatically sets the horizontal scale of the Gain graph.

Conditions

Measurement view: Gain

Group

Display commands

Syntax

```
DISPlay:NOISe:GAIN:X:AUTO
```

Arguments

None

DISPlay:NOISe:GAIN:X:OFFSet

Sets or queries the center frequency (offset) in the Gain display.

Conditions

Measurement view: Gain

Group

Display commands

Syntax

```
DISPlay:NOISe:GAIN:X:OFFSet <NRf>
DISPlay:NOISe:GAIN:X:OFFSet?
```

Arguments

`<NRf>` is the center frequency (offset) in the Gain display.

Range: -1.9900 GHz to +5.0000 GHz

Returns

`<NR3>` = a floating point value with an exponent, which is the center frequency (offset) value.

Examples

`DISPLAY:NOISE:GAIN:X:OFFSET?` might return `2.0000000009E+9`, indicating that the center frequency (offset) value is 2.0000 GHz.

DISPlay:NOISe:GAIN:X:SCALe

Sets or queries the horizontal scale of the Gain display.

Conditions

Measurement view: Gain

Group

Display commands

Syntax

```
DISPlay:NOISe:GAIN:X:SCALe <NRf>
DISPlay:NOISe:GAIN:X:SCALe
```

Related Commands

DISPlay:NOISe:GAIN:Y:SCALe (on page 803)

Arguments

<NRf> is the horizontal scale of the Gain display.

Range: 100 Hz to 6.2 GHz

Returns

<NR3> = a floating point value with an exponent, which is the horizontal scale value.

Examples

DISP:NOIS:GAIN:X:SCAL? might return 2.9900000000E+9, indicating that the horizontal scale value is 3.0 GHz.

DISPlay:NOISe:GAIN:Y:AUTO (No Query Form)

Automatically sets the vertical scale of the Gain graph.

Conditions

Measurement view: Gain

Group

Display commands

Syntax

```
DISPlay:NOISe:GAIN:Y:AUTO
```

Related Commands

DISPlay:NOISe:GAIN:Y:AUTO:STATe (on page 801)

DISPlay:NOISe:GAIN:X:AUTO (on page 800)

Arguments

None

DISPlay:NOISe:GAIN:Y:AUTO:STATe

Sets or queries the state of the automatic vertical scale and position of the Gain graph to on or off.

Conditions

Measurement view: Gain

Group

Display commands

Syntax

```
DISPlay:NOISe:GAIN:Y:AUTO:STATe { OFF | ON | 0 | 1 }
DISPlay:NOISe:GAIN:Y:AUTO:STATe?
```

Related Commands

DISPlay:NOISe:GAIN:Y:AUTO (on page 801)

Arguments

`OFF` or 0 sets the analyzer to manual setting of the vertical scale and position of the graph.

`ON` or 1 sets the analyzer to automatically set the vertical scale and position of the graph.

Returns

OFF or 0 means the analyzer is in manual mode for setting of the vertical scale and position of the graph.

ON or 1 means the analyzer is in automatic mode for setting of the vertical scale and position of the graph.

Examples

`DISPLAY:NOISE:GAIN:Y:AUTO:STATE ON` sets the analyzer to automatically set the vertical scale and position of the graph.

DISPlay:NOISe:GAIN:Y:PDIVision

Sets or queries the distance between graticule lines on the Gain display in dB.

Conditions

Measurement view: Gain

Group

Display commands

Syntax

```
DISPlay:NOISe:GAIN:Y:PDIVision <NRf>
DISPlay:NOISe:GAIN:Y:PDIVision?
```

Related Commands

Arguments

`<NRf>` is the distance between graticule lines on the Gain display.

Returns

`<NR2>` = a floating point value without exponents, which is the distance between graticule lines on the display in dB.

Examples

`DISPLAY:NOISE:GAIN:Y:PDIVISION?` might return 1.5000000000, indicating that the distance between graticule lines on the display is 1.50 dB.

DISPlay:NOISe:GAIN:Y:POSition

Sets or queries the vertical position of the Gain display.

Conditions

Measurement view: Gain

Group

Display commands

Syntax

```
DISPlay:NOISe:GAIN:Y:POSition <NRf>
DISPlay:NOISe:GAIN:Y:POSition?
```

Related Commands

Arguments

<NRf> is the vertical position of the Gain display.

Range: 5.12 to 26.5

Returns

<NR2> = a floating point value without exponents, which is the vertical position value.

Examples

DISPLAY:NOISE:GAIN:Y:POSITION? might return 26.5000000000, indicating that the vertical position is 26.5.

DISPlay:NOISe:GAIN:Y:SCALe

Sets or queries the vertical scale of the Gain display.

Conditions

Measurement view: Gain

Group

Display commands

Syntax

```
DISPlay:NOISe:GAIN:Y:SCALe <NRf>
DISPlay:NOISe:GAIN:Y:SCALe?
```

Related Commands

Arguments

<NRf> is the vertical scale of the Gain display.

Range: 10.2 to 31.6

Returns

<NR2> = a floating point value without an exponent, which is the vertical scale value.

Examples

`DISPLAY:NOISE:GAIN:Y:SCALE?` might return `10.0000000000`, indicating that the vertical scale is 10.0 dB

DISPlay:NOISe:MEASview:DELete (No Query Form)

Deletes the selected Noise and Gain measurement view.

Conditions

Measurement views: All Noise and Gain displays

Group

Display commands

Syntax

```
DISPlay:NOISe:MEASview:DELete { GAIN | YFACtor | FIGure | TEMPerature | TABLe | UNCertainty }
```

Related Commands

DISPlay:NOISe:MEASview:SELect (on page 805)

DISPlay:NOISe:MEASview:NEW (on page 804)

Arguments

`GAIN`: Deletes the Gain display from the view.

`YFACtor`: Deletes the Y Factor display from the view.

`FIGure`: Deletes the Noise Figure display from the view.

`TEMPerature`: Deletes the Noise Temperature display from the view.

`TABLe`: Deletes the Noise Table display from the view.

`UNCertainty`: Deletes the Uncertainty Calculator display from the view.

Examples

`DISPLAY:NOISE:MEASVIEW:DELETE TABL` will delete the Noise Table display view from the screen.

DISPlay:NOISe:MEASview:NEW (No Query Form)

Displays a new Noise and Gain measurement view.

Conditions

Measurement views: All Noise and Gain displays

Group

Display commands

Syntax

```
DISPlay:NOISe:MEASview:NEW { GAIN | YFACtor | FIGure | TEMPerature | TABLe | UNCertainty }
```

Related Commands

DISPlay:NOISe:MEASview:SELect (on page 805)

DISPlay:NOISe:MEASview:DELete (on page 804)

Arguments

GAIN: Displays the Gain display.

YFACtor: Displays the Y Factor display.

FIGure: Displays the Noise Figure display.

TEMPerature: Displays the Noise Temperature display.

TABLe: Displays the Noise Table display.

UNCertainty: Displays the Uncertainty Calculator display.

Examples

DISPLAY:NOISE:MEASVIEW:NEW YFAC will bring the Y Factor display up on the screen.

DISPlay:NOISe:MEASview:SELect

Selects a noise measurement view. The query command returns the currently selected view.

Selecting a measurement optimizes it. Other measurements may be optimized as a side effect. Refer to the DISplay:WINDow:OPTImized:MEASurement query.

Conditions

Measurement views: All Noise and Gain displays The measurement view must be displayed.

Group

Display commands

Syntax

```
DISPlay:NOISe:MEASview:SELect { GAIN | YFACtor | FIGure | TEMPerature | TABLe | UNCertainty }
DISPlay:NOISe:MEASview:SELect?
```

Related Commands

DISPlay:NOISe:MEASview:DELete (on page 804)

DISPlay:NOISe:MEASview:NEW (on page 804)

Arguments

GAIN: Selects the Gain display.

YFACtor: Selects the Y Factor display.

FIGure: Selects the Noise Figure display.

TEMPerature: Selects the Noise Temperature display.

TABLe: Selects the Noise Table display.

UNCertainty: Selects the Uncertainty Calculator display.

Returns

GAIN: Means that the Gain display is selected.

YFAC: Means that the Y Factor display is selected.

FIG: Means that the Noise Figure display is selected.

TEMP: Means that the Noise Temperature display is selected.

TABL: Means that the Noise Table display is selected.

UNC: Means that the Uncertainty Calculator display is selected.

Examples

DISPLAY:NOISE:MEASVIEW:SELECT GAIN will select the Gain display.

DISPlay:NOISe:POWer:LINear:STATe

Sets or queries whether the linear units are shown (on) or not shown (off) on the Noise Table display.

Conditions

Measurement view: Noise Table

Group

Display commands

Syntax

```
DISPlay:NOISe:POWer:LINear:STATe { OFF | ON | 0 | 1 }
DISPlay:NOISe:POWer:LINear:STATe?
```

Arguments

OFF or 0 sets the analyzer to show the linear units on the display.

ON or 1 sets the analyzer to not show the linear units on the display.

Returns

OFF or 0 indicates that the analyzer is set to show the linear units on the display.

ON or 1 indicates that the analyzer is set to not show the linear units on the display.

Examples

DISPLAY:NOISE:POWER:LINEAR:STATE? might return 1, indicating that linear units are not showing on the Noise Table display.

DISPlay:NOISe:TEMPerature[:SCALe]:AUTO (No Query Form)

Automatically scales the Noise Temperature graph.

Conditions

Measurement view: Noise Temperature

Group

Display commands

Syntax

```
DISPlay:NOISe:TEMPerature[:SCALe]:AUTO
```

Arguments

None

DISPlay:NOISe:TEMPerature:WINDow:TRACe:GRATicule:GRID:STATe

Sets or queries the graticule state in the Noise Temperature display to on (showing) or off (hidden).

Conditions

Measurement view: Noise Temperature

Group

Display commands

Syntax

```
DISPlay:NOISe:TEMPerature:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:NOISe:TEMPerature:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or 0 hides the graticule in the Noise Temperature display.

`ON` or 1 shows the graticule in the Noise Temperature display.

Returns

0 means the graticule in the Noise Temperature display is hidden (off).

1 means the graticule in the Noise Temperature display is showing (on).

Examples

`DISPLAY:NOISE:TEMPERATURE:WINDOW:TRACE:GRATICULE:GRID:STATE ON` will turn on the graticule in the Noise Temperature display.

`DISPLAY:NOISE:TEMPERATURE:WINDOW:TRACE:GRATICULE:GRID:STATE?` might return `1`, indicating that the graticule in the Noise Temperature display is showing (on).

DISPlay:NOISe:TEMPerature:WINDow:TRACe:LEGend:STATe

Sets or queries the trace legend state in the Noise Temperature display to on (showing) or off (hidden).

Conditions

Measurement view: Noise Temperature

Group

Display commands

Syntax

```
DISPlay:NOISe:TEMPerature:WINDow:TRACe:LEGend:STATe { OFF | ON | 0 | 1 }
DISPlay:NOISe:TEMPerature:WINDow:TRACe:LEGend:STATe?
```

Arguments

`OFF` or 0 hides the trace legend in the Noise Temperature display.

`ON` or 1 shows the trace legend in the Noise Temperature display.

Returns

0 means the trace legend in the Noise Temperature display is hidden (off).

1 means the trace legend in the Noise Temperature display is showing (on).

Examples

`DISPLAY:NOISE:TEMPERATURE:WINDOW:TRACE:LEGEND:STATE ON` will turn on the trace legend in the Noise Temperature display.

`DISPLAY:NOISE:TEMPERATURE:WINDOW:TRACE:LEGEND:STATE? 1` indicates that the trace legend in the Noise Temperature display is showing (on).

DISPlay:NOISe:TEMPerature:WINDow:TRACe:MEASPoints:STATe

Sets or queries the measurement points state in the Noise Temperature display to on (showing) or off (hidden).

Conditions

Measurement view: Noise Temperature

Group

Display commands

Syntax

```
DISPlay:NOISe:TEMPerature:WINDow:TRACe:MEASPoints:STATe { OFF | ON | 0 | 1 }
DISPlay:NOISe:TEMPerature:WINDow:TRACe:MEASPoints:STATe?
```

Arguments

`OFF` or 0 hides the measurement points in the Noise Temperature display.

`ON` or 1 shows the measurement points in the Noise Temperature display.

Returns

0 means the measurement points in the Noise Temperature display is hidden (off).

1 means the measurement points in the Noise Temperature display is showing (on).

Examples

`DISPLAY:NOISE:TEMPERATURE:WINDOW:TRACE:MEASPOINTS:STATE ON` will turn on the measurement points in the Noise Temperature display.

`DISPLAY:NOISE:TEMPERATURE:WINDOW:TRACE:MEASPOINTS:STATE?` might return `1`, indicating that the measurement points in the Noise Temperature display are showing (on).

DISPlay:NOISe:TEMPerature:X:AUTO (No Query Form)

Automatically sets the horizontal scale of the Noise Temperature graph.

Conditions

Measurement view: Noise Temperature

Group

Display commands

Syntax

```
DISPlay:NOISe:TEMPerature:X:AUTO
```

Arguments

None

DISPlay:NOISe:TEMPerature:X:OFFSet

Sets or queries the center frequency (offset) value for the Noise Temperature display.

Conditions

Measurement view: Noise Temperature

Group

Display commands

Syntax

```
DISPlay:NOISe:TEMPerature:X:OFFSet <NRf>
DISPlay:NOISe:TEMPerature:X:OFFSet?
```

Arguments

<NRf> is the center frequency (offset) value.

Range: -1.9900 GHz to +5.0000 GHz

Returns

<NR3> = a floating point value with an exponent, which is the center frequency (offset) value.

Examples

DISPLAY:NOISE:TEMPERATURE:X:OFFSET? might return 2.0000000009E+9, indicating that the center frequency (offset) value is 2.0000 GHz.

DISPlay:NOISe:TEMPerature:X[:SCALe]

Sets or queries the horizontal scale of the Noise Temperature display.

Conditions

Measurement view: Noise Temperature

Group

Display commands

Syntax

```
DISPlay:NOISe:TEMPerature:X[:SCALe] <NRf>
DISPlay:NOISe:TEMPerature:X[:SCALe]
```

Related Commands

DISPlay:NOISe:GAIN:Y:SCALe (on page 803)

Arguments

<NRf> is the horizontal scale of the Noise Temperature display.

Range: 100 Hz to 6.2 GHz

Returns

<NR3> = a floating point value with an exponent, which is the horizontal scale value.

Examples

DISP:NOIS:TEMP:X:SCAL? might return 2.9900000000E+9, indicating that the horizontal scale value is 3.0 GHz.

DISPlay:NOISe:TEMPerature:X:STARt

Sets or queries the Start Frequency value in the Noise Temperature display.

Conditions

Measurement views: Noise Temperature

Group

Display commands

Syntax

```
DISPlay:NOISe:TEMPerature:X:STARt <NRf>
DISPlay:NOISe:TEMPerature:X:STARt?
```

Arguments

<NRf> is the Start Frequency value.

Returns

<NR3> = floating point value with an exponent, which is the Start Frequency value.

Examples

DISPLAY:NOISE:TEMPERATURE:X:START 1.5GHz sets the Start Frequency in the Noise Temperature display to 1.5000 GHz.

DISPLAY:NOISE:TEMPERATURE:X:START? might return 1.5000000000E+9, indicating that the Start Frequency in the Noise Temperature display is 1.5000 GHz.

DISPlay:NOISe:TEMPerature:X:STOP

Sets or queries the Stop Frequency value in the Noise Temperature display.

Conditions

Measurement views: Noise Temperature

Group

Display commands

Syntax

```
DISPlay:NOISe:TEMPerature:X:STOP <NRf>
DISPlay:NOISe:TEMPerature:X:STOP?
```

Arguments

<NRf> is the Stop Frequency value.

Returns

<NR3> = floating point value with an exponent, which is the Stop Frequency value.

Examples

`DISPLAY:NOISE:TEMPERATURE:X:STOP 1.5GHz` sets the Stop Frequency in the Noise Temperature display to 1.5000 GHz.

`DISPLAY:NOISE:TEMPERATURE:X:STOP?` might return `1.5000000000E+9`, indicating that the Stop Frequency in the Noise Temperature display is 1.5000 GHz.

DISPlay:NOISe:TEMPerature:Y:AUTO (No Query Form)

Automatically sets the vertical scale of the Noise Temperature graph.

Conditions

Measurement view: Noise Temperature

Group

Display commands

Syntax

```
DISPlay:NOISe:TEMPerature:Y:AUTO
```

Arguments

None

DISPlay:NOISe:TEMPerature:Y:AUTO:STATe

Sets or queries the state of the automatic vertical scale and position of the graph to on or off.

Conditions

Measurement view: Noise Temperature

Group

Display commands

Syntax

```
DISPlay:NOISe:TEMPerature:Y:AUTO:STATe { OFF | ON | 0 | 1 }
DISPlay:NOISe:TEMPerature:Y:AUTO:STATe?
```

Arguments

`OFF` or 0 sets the analyzer to manual setting of the vertical scale and position of the graph.

`ON` or 1 sets the analyzer to automatically set the vertical scale and position of the graph.

Returns

`OFF` or 0 means the analyzer is in manual mode for setting of the vertical scale and position of the graph.

`ON` or 1 means the analyzer is in automatic mode for setting of the vertical scale and position of the graph.

Examples

`DISPLAY:NOISE:TEMPERATURE:Y:AUTO:STATE ON` sets the analyzer to automatically set the vertical scale and position of the graph.

DISPlay:NOISe:TEMPerature:Y:PDIVision

Sets or queries the distance between graticule lines on the display in dB.

Conditions

Measurement view: Noise Temperature

Group

Display commands

Syntax

```
DISPlay:NOISe:TEMPerature:Y:PDIVision <NRf>
DISPlay:NOISe:TEMPerature:Y:PDIVision?
```

Arguments

<NRf> is the distance between graticule lines on the display.

Range: 100 mdB to 5.00 dB

Returns

<NR2> = a floating point value without an exponent, which is the distance between graticule lines on the display in dB.

Examples

DISPLAY:NOISE:TEMPERATURE:Y:PDIVISION? might return 1.5000000000, indicating that the distance between graticule lines on the display is 1.50 dB.

DISPlay:NOISe:TEMPerature:Y:POSition

Sets or queries the vertical position of the Noise Temperature display.

Conditions

Measurement view: Noise Temperature

Group

Display commands

Syntax

```
DISPlay:NOISe:TEMPerature:Y:POSition <NRf>
DISPlay:NOISe:TEMPerature:Y:POSition?
```

Arguments

<NRf> is the vertical position of the Noise Temperature display.

Range: 5.12 to 26.5

Returns

<NR2> = a floating point value without an exponent, which is the vertical position value.

The vertical position value

Examples

DISPLAY:NOISE:TEMPERATURE:Y:POSITION? might return 26.5000000000, indicating that the vertical position is 26.5.

DISPlay:NOISe:TEMPerature:Y:SCALe

Sets or queries the vertical scale of the Noise Temperature display.

Conditions

Measurement view: Noise Temperature

Group

Display commands

Syntax

```
DISPlay:NOISe:TEMPerature:Y:SCALe <NRf>
DISPlay:NOISe:TEMPerature:Y:SCALe?
```

Arguments

<NRf> is the vertical scale of the Noise Temperature display.

Range: 10.2 to 31.6

Returns

<NR2> = a floating point value without an exponent, which is the vertical scale value.

Examples

DISPLAY:NOISE:TEMPERATURE:Y:SCALE? might return 10.0000000000, indicating that the vertical scale is 10.0 dB.

DISPlay:NOISe:UNCERTainty:RESult:GAIN? (Query Only)

Queries the gain computed uncertainty value (dB).

Conditions

Measurement view: Uncertainty Calculator

Group

Display commands

Syntax

```
DISPlay:NOISe:UNCERTainty:RESult:GAIN?
```

Related Commands

DISPlay:NOISe:UNCERTainty:RESult:NFIGure? (on page 813)

Returns

<NR3> = a floating point value with an exponent, which is the gain computed uncertainty value (dB).

Examples

DISPLAY:NOISE:UNCERTAINTY:RESULT:GAIN? might return 783.0000000000E-3, indicating that the computed uncertainty value for gain is 0.783 dB.

DISPlay:NOISe:UNCERTainty:RESult:NFIGure? (Query Only)

Queries the noise figure computed uncertainty value (dB).

Conditions

Measurement view: Uncertainty Calculator

Group

Display commands

Syntax

```
DISPlay:NOISe:UNCERTainty:RESult:NFIGure?
```

Related Commands

DISPlay:NOISe:UNCERTainty:RESult:GAIN? (on page 813)

Returns

<NR3> = a floating point value with an exponent, which is the noise figure computed uncertainty value (dB).

Examples

`DISPLAY:NOISE:UNCERTAINTY:RESULT:NFIGURE?` might return `783.0000000000E-3`, indicating that the computed uncertainty value for noise figure is 0.783 dB.

DISPlay:NOISe:YFACtor[:SCALe]:AUTO (No Query Form)

Automatically scales the Y Factor graph.

Conditions

Measurement view: Y Factor

Group

Display commands

Syntax

```
DISPlay:NOISe:YFACtor[:SCALe]:AUTO
```

Arguments

None

DISPlay:NOISe:YFACtor:WINDow:TRACe:GRATicule:GRID:STATe

Sets or queries the graticule state in the Y Factor display to on (showing) or off (hidden).

Conditions

Measurement view: Y Factor

Group

Display commands

Syntax

```
DISPlay:NOISe:YFACtor:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:NOISe:YFACtor:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

OFF or 0 hides the graticule in the Y Factor display.

ON or 1 shows the graticule in the Y Factor display.

Returns

0 means the graticule in the Y Factor display is hidden (off).

1 means the graticule in the Y Factor display is showing (on).

Examples

DISPLAY:NOISE:YFACTOR:WINDOW:TRACE:GRATICULE:GRID:STATE ON will turn on the graticule in the Y Factor display.

DISPLAY:NOISE:YFACTOR:WINDOW:TRACE:GRATICULE:GRID:STATE? might return 1, indicating that the graticule in the Y Factor display is showing (on).

DISPlay:NOISe:YFACtor:WINDow:TRACe:LEGend:STATe

Sets or queries the trace legend state in the Y Factor display to on (showing) or off (hidden).

Conditions

Measurement view: Y Factor

Group

Display commands

Syntax

```
DISPlay:NOISe:YFACtor:WINDow:TRACe:LEGend:STATe { OFF | ON | 0 | 1 }
DISPlay:NOISe:YFACtor:WINDow:TRACe:LEGend:STATe?
```

Arguments

OFF or 0 hides the trace legend in the Y Factor display.

ON or 1 shows the trace legend in the Y Factor display.

Returns

0 means the trace legend in the Y Factor display is hidden (off).

1 means the trace legend in the Y Factor display is showing (on).

Examples

DISPLAY:NOISE:YFACTOR:WINDOW:TRACE:LEGEND:STATE ON will turn on the trace legend in the Y Factor display.

DISPLAY:NOISE:YFACTOR:WINDOW:TRACE:LEGEND:STATE? might return 1 indicates that the trace legend in the Y Factor display is showing (on).

DISPlay:NOISe:YFACtor:WINDow:TRACe:MEASPoints:STATe

Sets or queries the measurement points state in the Y Factor display to on (showing) or off (hidden).

Conditions

Measurement view: Y Factor

Group

Display commands

Syntax

```
DISPlay:NOISe:YFACtor:WINDow:TRACe:MEASPoints:STATe { OFF | ON | 0 | 1 }
DISPlay:NOISe:YFACtor:WINDow:TRACe:MEASPoints:STATe?
```

Arguments

`OFF` or 0 hides the measurement points in the Y Factor display.

`ON` or 1 shows the measurement points in the Y Factor display.

Returns

0 means the measurement points in the Y Factor display is hidden (off).

1 means the measurement points in the Y Factor display is showing (on).

Examples

`DISPLAY:NOISE:YFACTOR:WINDOW:TRACE:MEASPOINTS:STATE ON` will turn on the measurement points in the Y Factor display.

`DISPLAY:NOISE:YFACTOR:WINDOW:TRACE:MEASPOINTS:STATE?` might return 1, indicating that the measurement points in the Y Factor display are showing (on).

DISPlay:NOISe:YFACtor:X:AUTO (No Query Form)

Automatically sets the horizontal scale of the Y Factor graph.

Conditions

Measurement view: Y Factor

Group

Display commands

Syntax

```
DISPlay:NOISe:YFACtor:X:AUTO
```

Arguments

None

DISPlay:NOISe:YFACtor:X:OFFSet

Sets or queries the center frequency (offset) in the Y Factor display.

Conditions

Measurement view: Y Factor

Group

Display commands

Syntax

```
DISPlay:NOISe:YFACtor:X:OFFSet <NRf>
DISPlay:NOISe:YFACtor:X:OFFSet?
```

Arguments

<NRf> is the center frequency (offset) in the Y Factor display.

Range: -1.9900 GHz to +5.0000 GHz

Returns

<NR3> = a floating point value with an exponent, which is the center frequency (offset) value.

Examples

DISPLAY:NOISE:YFACTOR:X:OFFSET? might return 2.0000000009E+9, indicating that the center frequency (offset) value is 2.0000 GHz.

DISPlay:NOISe:YFACtor:X[:SCALe]

Sets or queries the horizontal scale of the Y Factor display.

Conditions

Measurement view: Y Factor

Group

Display commands

Syntax

```
DISPlay:NOISe:YFACtor:X[:SCALe] <NRf>
DISPlay:NOISe:YFACtor:X[:SCALe]
```

Related Commands

DISPlay:NOISe:GAIN:Y:SCALe (on page 803)

Arguments

<NRf> is the horizontal scale of the Y Factor display.

Range: 100 Hz to 6.2 GHz

Returns

<NR3> = a floating point value with an exponent, which is the horizontal scale value.

Examples

DISP:NOIS:YFAC:X:SCAL? might return 2.9900000000E+9, indicating that the horizontal scale value is 3.0 GHz.

DISPlay:NOISe:YFACtor:X:STARt

Sets or queries the Start Frequency value in the Y Factor display.

Conditions

Measurement views: Y Factor

Group

Display commands

Syntax

```
DISPlay:NOISe:YFACtor:X:STARt <NRf>
DISPlay:NOISe:YFACtor:X:STARt?
```

Arguments

<NRf> is the Start Frequency value in the Y Factor display.

Returns

<NR3> = floating point value with an exponent, which is the Start Frequency value.

Examples

DISPLAY:NOISE:YFACTOR:X:START 1.5GHz sets the Start Frequency in the Y Factor display to 1.5000 GHz.

DISPLAY:NOISE:YFACTOR:X:START? might return 1.5000000000E+9, indicating that the Start Frequency in the Y Factor display is 1.5000 GHz.

DISPlay:NOISe:YFACtor:X:STOP

Sets or queries the Stop Frequency value Y Factor display.

Conditions

Measurement views: Y Factor

Group

Display commands

Syntax

```
DISPlay:NOISe:YFACtor:X:STOP <NRf>
DISPlay:NOISe:YFACtor:X:STOP?
```

Arguments

<NRf> is the Stop Frequency value in the Y Factor display.

Returns

<NR3> = floating point value with an exponent, which is the Stop Frequency value.

Examples

DISPLAY:NOISE:YFACTOR:X:STOP 1.5GHz sets the Stop Frequency in the Y Factor display to 1.5000 GHz.

DISPLAY:NOISE:YFACTOR:X:STOP? might return 1.5000000000E+9, indicating that the Stop Frequency in the Y Factor display is 1.5000 GHz.

DISPlay:NOISe:YFACtor:Y:AUTO (No Query Form)

Automatically sets the vertical scale of the Y Factor graph.

Conditions

Measurement view: Y Factor

Group

Display commands

Syntax

DISPlay:NOISe:YFACtor:Y:AUTO

Arguments

None

DISPlay:NOISe:YFACtor:Y:AUTO:STATe

Sets or queries the state of the automatic vertical scale and position of the Y Factor graph to on or off.

Conditions

Measurement view: Y Factor

Group

Display commands

Syntax

```
DISPlay:NOISe:YFACtor:Y:AUTO:STATe { OFF | ON | 0 | 1 }
DISPlay:NOISe:YFACtor:Y:AUTO:STATe?
```

Arguments

OFF or 0 sets the analyzer to manual setting of the vertical scale and position of the graph.

ON or 1 sets the analyzer to automatically set the vertical scale and position of the graph.

Returns

OFF or 0 means the analyzer is in manual mode for setting of the vertical scale and position of the graph.

ON or 1 means the analyzer is in automatic mode for setting of the vertical scale and position of the graph.

Examples

DISPLAY:NOISE:YFACTOR:Y:AUTO:STATE ON sets the analyzer to automatically set the vertical scale and position of the graph.

DISPlay:NOISe:YFACtor:Y:PDIVision

Sets or queries the distance between graticule lines on the display in dB.

Conditions

Measurement view: Y Factor

Group

Display commands

Syntax

```
DISPlay:NOISe:YFACtor:Y:PDIVision <NRf>
DISPlay:NOISe:YFACtor:Y:PDIVision?
```

Arguments

<NRf> is the distance between graticule lines on the display in dB. Range: 100 mdB to 5.00 dB

Returns

<NR2> = a floating point value without an exponent, which is the distance between graticule lines on the display in dB.

Examples

`DISPLAY:NOISE:YFACTOR:Y:PDIVISION?` might return `1.5000000000`, indicating that the distance between graticule lines on the display is 1.50 dB.

DISPlay:NOISe:YFACtor:Y:POSition

Sets or queries the vertical position of the Y Factor display.

Conditions

Measurement view: Y Factor

Group

Display commands

Syntax

```
DISPlay:NOISe:YFACtor:Y:POSition <NRf>
DISPlay:NOISe:YFACtor:Y:POSition?
```

Arguments

`<NRf>` is the vertical position of the Y Factor display.

Range: 5.12 to 26.5

Returns

`<NR2>` = a floating point value without an exponent, which is the vertical position value.

Examples

`DISPLAY:NOISE:YFACTOR:Y:POSITION?` might return `26.5000000000`, indicating that the vertical position is 26.5.

DISPlay:NOISe:YFACtor:Y:SCALe

Sets or queries the vertical scale of the Y Factor display.

Conditions

Measurement view: Y Factor

Group

Display commands

Syntax

```
DISPlay:NOISe:YFACtor:Y:SCALe <NRf>
DISPlay:NOISe:YFACtor:Y:SCALe?
```

Related Commands

Arguments

`<NRf>` is the vertical scale of the Y Factor display.

Range: 10.2 to 31.6 dB

Returns

`<NR2>` = a floating point value without an exponent, which is the vertical scale

value.

Examples

`DISPLAY:NOISE:YFACTOR:Y:SCALE?` might return `10.0000000000`, indicating that the vertical scale is 10.0 dB.

DISPlay:OBWidth:MARKer:SHOW:STATe

Determines whether to show or hide the readout for the selected marker in the Occupied Bandwidth view.

Conditions

Measurement views: Occupied Bandwidth

Group

Display commands

Syntax

```
DISPlay:OBWidth:MARKer:SHOW:STATe { OFF | ON | 0 | 1 }
DISPlay:OBWidth:MARKer:SHOW:STATe?
```

Arguments

`OFF` or 0 hides the readout for the selected marker in the view.

`ON` or 1 shows the readout for the selected marker in the view.

Examples

`DISPLAY:OBWIDTH:MARKER:SHOW:STATE ON` shows the readout for the selected marker in the view.

DISPlay:OBWidth:RESet:SCALe (No Query Form)

Resets the horizontal and vertical scale to the default values described below in the Occupied Bandwidth view.

Vertical offset = Reference level, Vertical scale = 100 dB,

Horizontal offset = Center frequency, and Horizontal scale = Default span

Conditions

Measurement views: Occupied Bandwidth

Group

Display commands

Syntax

```
DISPlay:OBWidth:RESet:SCALe
```

Arguments

None

Examples

`DISPLAY:OBWIDTH:RESET:SCALE` resets the horizontal and vertical scale to the default values.

DISPlay:OBWidth:SELected:BANDwidth

Sets or queries the bandwidth (OBW or x dB BW) to measure in the Occupied Bandwidth view.

Conditions

Measurement views: Occupied Bandwidth

Group

Display commands

Syntax

```
DISPlay:OBWidth:SELected:BANDwidth { OBWidth | XDBbandwidth }
DISPlay:OBWidth:SELected:BANDwidth?
```

Arguments

`OBWidth` selects the occupied bandwidth to measure.

`XDBbandwidth` selects the x dB bandwidth to measure.

Examples

`DISPLAY:OBWIDTH:SELECTED:BANDWIDTH OBWidth` selects the occupied bandwidth to measure.

DISPlay:OBWidth:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions

Measurement views: Occupied Bandwidth

Group

Display commands

Syntax

```
DISPlay:OBWidth:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:OBWidth:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or 0 hides the graticule grid.

`ON` or 1 shows the graticule grid.

Examples

`DISPLAY:OBWIDTH:WINDOW:TRACE:GRATICULE:GRID:STATE ON` shows the graticule grid on the screen.

DISPlay:OBWidth:X[:SCALe]

Sets or queries the horizontal range of the Occupied Bandwidth view.

Conditions

Measurement views: Occupied Bandwidth

Group

Display commands

Syntax

```
DISPlay:OBWidth:X[:SCALe] <value>
DISPlay:OBWidth:X[:SCALe]?
```

Related Commands

DISPlay:OBWidth:X[:SCALe]:OFFSet (on page 823)

Arguments

`<value>::<NRf>` specifies the horizontal range.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples

`DISPLAY:OBWIDTH:X:SCALE 10MHz` sets the horizontal range to 10 MHz.

DISPlay:OBWidth:X[:SCALe]:AUTO (No Query Form)

Rescales the horizontal axis automatically to fit the waveform to the screen in the Occupied Bandwidth view.

Conditions

Measurement views: Occupied Bandwidth

Group

Display commands

Syntax

```
DISPlay:OBWidth:X[:SCALe]:AUTO
```

Arguments

None

Examples

`DISPLAY:OBWIDTH:X:SCALE:AUTO` rescales the horizontal scale automatically to fit the waveform to the screen.

DISPlay:OBWidth:X[:SCALe]:OFFSet

Sets or queries the minimum horizontal value (left edge) of the Occupied Bandwidth view.

Conditions

Measurement views: Occupied Bandwidth

Group

Display commands

Syntax

```
DISPlay:OBWidth:X[:SCALe]:OFFSet <value>
DISPlay:OBWidth:X[:SCALe]:OFFSet?
```

Related Commands

DISPlay:OBWidth:X[:SCALe] (on page 822)

Arguments

`<value>::<NRf>` specifies the minimum horizontal value.

Range: [(center frequency) - (X scale) × 0.9] to [(center frequency) + (X scale) × 0.9]

Examples

`DISPLAY:OBWIDTH:X:SCALE:OFFSET 1.45GHz` sets the minimum horizontal value to 1.45 GHz in the Occupied Bandwidth view.

DISPlay:OBWidth:Y[:SCALe]

Sets or queries the vertical range of the Occupied Bandwidth view.

Conditions

Measurement views: Occupied Bandwidth

Group

Display commands

Syntax

```
DISPlay:OBWidth:Y[:SCALe] <value>
DISPlay:OBWidth:Y[:SCALe]?
```

Related Commands

DISPlay:OBWidth:Y[:SCALe]:OFFSet (on page 825)

Arguments

`<value>::<NRf>` specifies the vertical range. Range: 0.1 to 200 dB.

Examples

`DISPLAY:OBWIDTH:Y:SCALE 100` sets the vertical range to 100 dB in the Occupied Bandwidth view.

DISPlay:OBWidth:Y[:SCALe]:AUTO (No Query Form)

Rescales the vertical axis automatically to fit the waveform to the screen in the Occupied Bandwidth view.

Conditions

Measurement views: Occupied Bandwidth

Group

Display commands

Syntax

```
DISPlay:OBWidth:Y[:SCALe]:AUTO
```

Arguments

None

Examples

`DISPLAY:OBWIDTH:Y:SCALE:AUTO` rescales the vertical scale automatically to fit the waveform to the screen.

DISPlay:OBWidth:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (the value at the top edge of the vertical axis) in the Occupied Bandwidth view.

Conditions

Measurement views: Occupied Bandwidth

Group

Display commands

Syntax

```
DISPlay:OBWidth:Y[:SCALe]:OFFSet <value>
DISPlay:OBWidth:Y[:SCALe]:OFFSet?
```

Related Commands

DISPlay:OBWidth:Y[:SCALe] (on page 824)

Arguments

`<value>::<NRf>` specifies the vertical offset. Range: -170 to +50 dBm.

Examples

`DISPLAY:OBWIDTH:Y:SCALE:OFFSET -12.5` sets the vertical offset to -12.5 dBm in the Occupied Bandwidth view.

DISPlay:OFDM:CONSte[:SCALe]

Sets or queries the scale (zoom) in the OFDM Constellation view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:CONSte[:SCALe] <value>
DISPlay:OFDM:CONSte[:SCALe]?
```

Arguments

`<value>::<NRf>` specifies the zoom setting in the OFDM Constellation view.

A positive value zooms out and a negative value zooms in.

Examples

`DISPLAY:OFDM:CONSTE[:SCALE] 5` zooms the display out by 5

DISPlay:OFDM:CONSte:X:OFFSet

Sets or queries the horizontal offset in the OFDM Constellation view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:CONSte:X:OFFSet <value>
DISPlay:OFDM:CONSte:X:OFFSet?
```

Arguments

`<value>::<NRf>` specifies the offset range.

Examples

`DISPLAY:OFDM:CONSTE:X:OFFSET 2.7` sets the horizontal offset to 2.7 units to the left.

DISPlay:OFDM:CONSte:Y:OFFSet

Sets or queries the vertical offset in the OFDM Constellation view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:CONSte:Y:OFFSet <value>
DISPlay:OFDM:CONSte:Y:OFFSet?
```

Arguments

`<value>::<NRf>` specifies the offset range.

A positive value moves the offset up; and negative value moves the offset down.

Examples

`DISPLAY:OFDM:CONSTE:Y:OFFSET 2` sets the vertical offset to up 2 units.

DISPlay:OFDM:CRESponse:FREQuency:AUTO (No Query Form)

Rescales the horizontal axis automatically to fit the waveform to the screen in the OFDM Channel response graph.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:CRESponse:FREQuency:AUTO
```

Arguments

None

Examples

`DISPLAY:OFDM:CRESPONSE:FREQUENCY:AUTO` rescales the horizontal scale automatically to fit the waveform to the screen.

DISPlay:OFDM:CRESponse:FREQuency:OFFSet

Sets or queries the frequency offset in the OFDM Channel response graph.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:CRESponse:FREQuency:OFFSet <value>
DISPlay:OFDM:CRESponse:FREQuency:OFFSet?
```

Arguments

`<value>::<NRf>` specifies the frequency offset.

Examples

`DISPLAY:OFDM:CRESPONSE:FREQUENCY:OFFSET 1.45E+9` sets the frequency offset to 1.45 GHz.

DISPlay:OFDM:CRESponse:FREQuency[:SCALe]

Sets or queries the horizontal range of the OFDM Channel response graph.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:CRESponse:FREQuency[:SCALe] <value>
DISPlay:OFDM:CRESponse:FREQuency[:SCALe]?
```

Arguments

`<value>::= <NRf>` specifies the horizontal range.

Examples

`DISPLAY:OFDM:CRESPONSE:FREQUENCY[:SCALE] 10E+6` sets the horizontal range to 10 MHz.

DISPlay:OFDM:CRESponse:MAGNitude:AUTO (No Query Form)

Rescales the magnitude automatically to fit the waveform to the screen in the OFDM Channel response graph.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:CRESponse:MAGNitude:AUTO
```

Arguments

None

Examples

`DISPLAY:OFDM:CRESPONSE:MAGNITUDE:AUTO` rescales the magnitude scale automatically to fit the waveform to the screen.

DISPlay:OFDM:CRESponse:MAGNitude:OFFSet

Sets or queries the magnitude offset in the OFDM Channel response graph.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:CRESponse:MAGNitude:OFFSet <value>
DISPlay:OFDM:CRESponse:MAGNitude:OFFSet?
```

Arguments

`<value>::= <NRf>` specifies the magnitude offset.

Examples

`DISPLAY:OFDM:CRESPONSE:MAGNITUDE:OFFSET 5` sets the magnitude offset to 5 dB.

DISPlay:OFDM:CRESponse:MAGNitude:PDIVision

Sets or queries the scale (per division) of the magnitude in the OFDM Channel response graph.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:CRESponse:MAGNitude:PDIVision <value>
DISPlay:OFDM:CRESponse:MAGNitude:PDIVision?
```

Arguments

`<value>::<NRf>` specifies the vertical scale (per division).

Examples

`DISPLAY:OFDM:CRESPONSE:MAGNITUDE:PDIVISION 2.7` sets the vertical scale to 2.70 dB/div.

DISPlay:OFDM:CRESponse:PHASe:AUTO (No Query Form)

Rescales the vertical axis automatically to fit the waveform to the screen in the OFDM Channel response graph.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:CRESponse:PHASe:AUTO
```

Arguments

None

Examples

`DISPLAY:OFDM:CRESPONSE:PHASE:AUTO` rescales the vertical scale automatically to fit the waveform to the screen.

DISPlay:OFDM:CRESponse:PHASe:OFFSet

Sets or queries the phase offset in the OFDM Channel response graph.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:CRESponse:PHASe:OFFSet <value>
DISPlay:OFDM:CRESponse:PHASe:OFFSet?
```

Arguments

`<value>::<NRf>` specifies the phase offset.

Examples

`DISPLAY:OFDM:CRESPONSE:PHASE:OFFSET -14.5` sets the minimum vertical value to -14.5 ° in the OFDM Channel response graph.

DISPlay:OFDM:CRESponse:PHASe[:SCALe]

Sets the vertical scale automatically to fit the waveform to the screen in the OFDM Channel response graph.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:CRESponse:PHASe[:SCALe] <value>
DISPlay:OFDM:CRESponse:PHASe[:SCALe]?
```

Arguments

`<value>::<NRf>` specifies the vertical range.

Examples

`DISPLAY:OFDM:CRESPONSE:PHASE[:SCALE] 30` sets the vertical range to 30 ° in the OFDM Channel response graph.

DISPlay:OFDM:CRESponse:WINDow:SELect:PLOT

Sets or queries the plot in the OFDM Channel Response window measurement.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:CRESponse:WINDow:SELect:PLOT { MAGNitude | PHASe }
DISPlay:OFDM:CRESponse:WINDow:SELect:PLOT?
```

Arguments

`MAGNitude` selects the magnitude display.

`PHASe` selects the phase display

Examples

`DISPLAY:OFDM:CRESPONSE:WINDOW:SELECT:PLOT PHASe` selects the phase display.

DISPlay:OFDM:CRESponse:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule in the OFDM Channel Response view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:CRESponse:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 1| 0}
DISPlay:OFDM:CRESponse:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or 0 hides the graticule in the graph.

`ON` or 1 shows the graticule in the graph.

Examples

`DISPLAY:OFDM:CRESPONSE:WINDOW:TRACE:GRATICULE:GRID:STATE OFF` turns off the graticule in the graph.

DISPlay:OFDM:EVM:FREQuency:AUTO (No Query Form)

Rescales the frequency automatically to fit the waveform to the screen in the OFDM EVM graph.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:EVM:FREQuency:AUTO
```

Arguments

None

Examples

`DISPLAY:OFDM:EVM:FREQUENCY:AUTO` rescales the frequency scale automatically to fit the waveform to the screen.

DISPlay:OFDM:EVM:FREQuency:OFFSet

Sets or queries the frequency offset in the OFDM EVM graph.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:EVM:FREQuency:OFFSet <value>
DISPlay:OFDM:EVM:FREQuency:OFFSet
```

Arguments

`<value>::<NRf>` specifies the frequency offset.

Examples

`DISPLAY:OFDM:EVM:FREQUENCY:OFFSET -21` sets the frequency offset to –21 subcarriers.

DISPlay:OFDM:EVM:FREQuency[:SCALe]

Sets or queries the horizontal range of the OFDM EVM graph.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:EVM:FREQuency[:SCALe] <value>
DISPlay:OFDM:EVM:FREQuency[:SCALe]?
```

Arguments

`<value>::= <NRf>` specifies the horizontal range.

Examples

`DISPLAY:OFDM:EVM:FREQUENCY[:SCALE] 10E+6` sets the horizontal range to 10 MHz.

DISPlay:OFDM:EVM:MARKer:SHOW:STATe

Determines whether to show or hide the readout for the selected marker in the OFDM EVM view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:EVM:MARKer:SHOW:STATe { OFF | ON | 1| 0 }
DISPlay:OFDM:EVM:MARKer:SHOW:STATe?
```

Arguments

`OFF` or 0 hides the readout for the selected marker in the graph.

`ON` or 1 shows the readout for the selected marker in the graph.

Examples

`DISPLAY:OFDM:EVM:MARKER:SHOW:STATE ON` shows the readout for the selected marker in the graph.

DISPlay:OFDM:EVM:TIME:AUTO (No Query Form)

Rescales the time automatically to fit the waveform to the screen in the OFDM EVM view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:EVM:TIME:AUTO
```

Arguments

None

Examples

`DISPLAY:OFDM:EVM:TIME:AUTO` rescales the time scale automatically to fit the waveform to the screen.

DISPlay:OFDM:EVM:TIME:OFFSet

Sets or queries the time offset in the OFDM EVM view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:EVM:TIME:OFFSet <value>
DISPlay:OFDM:EVM:TIME:OFFSet?
```

Arguments

`<value>::= <NRf>` specifies the magnitude offset.

Examples

`DISPLAY:OFDM:EVM:TIME:OFFSET -11.7` sets the time scale offset to −11.7 symbols.

DISPlay:OFDM:EVM:TIME[:SCALe]

Sets or queries the time scale in the OFDM EVM view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:EVM:TIME[:SCALe] <value>
DISPlay:OFDM:EVM:TIME[:SCALe]?
```

Arguments

`<value>::= <NRf>` specifies the time scale.

Examples

`DISPLAY:OFDM:EVM:TIME[:SCALE] 13` sets the time scale to 13 symbols.

DISPlay:OFDM:EVM:WINDow:SELect:PLOT

Sets or queries the plot in the OFDM EVM window measurement.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:EVM:WINDow:SELect:PLOT { BOTH | SCARrier | SYMBols }
DISPlay:OFDM:EVM:WINDow:SELect:PLOT?
```

Arguments

`SCARrier` selects the subcarrier display.

`SYMBols` selects the symbols display

`BOTH` selects both displays.

Examples

`DISPLAY:OFDM:EVM:WINDOW:SELECT:PLOT SYMBols` selects the symbols display.

DISPlay:OFDM:EVM:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule in the OFDM EVM view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:EVM:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 1| 0}
DISPlay:OFDM:EVM:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or 0 hides the graticule in the graph.

`ON` or 1 shows the graticule in the graph.

Examples

`DISPLAY:OFDM:EVM:WINDOW:TRACE:GRATICULE:GRID:STATE OFF` turns off the graticule in the graph.

DISPlay:OFDM:EVM:Y:AUTO (No Query Form)

Rescales the vertical scale automatically to fit the waveform in the OFDM EVM view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:EVM:Y:AUTO
```

Arguments

None

Examples

`DISPLAY:OFDM:EVM:Y:AUTO` rescales the vertical scale to fit the waveform on the screen.

DISPlay:OFDM:EVM:Y:OFFSet

Sets or queries the vertical offset in the OFDM EVM view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:EVM:Y:OFFSet <value>
DISPlay:OFDM:EVM:Y:OFFSet?
```

Arguments

`<value>::= <NRf>` specifies the vertical offset.

Examples

`DISPLAY:OFDM:EVM:Y:OFFSET -20` sets the vertical offset to –20%.

DISPlay:OFDM:EVM:Y[:SCALe]

Sets or queries the vertical range of the OFDM EVM view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:EVM:Y[:SCALe] <value>
DISPlay:OFDM:EVM:Y[:SCALe]?
```

Arguments

`<value>::= <NRf>` specifies the vertical scale.

Examples

`DISPLAY:OFDM:EVM:Y[:SCALE] 200` sets the vertical scale to 200%.

DISPlay:OFDM:FLATness:AUTO (No Query Form)

Automatically rescales the horizontal and vertical axes for the best display in the OFDM Spectral Flatness view.

Conditions

Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group

Display commands

Syntax

`DISPlay:OFDM:FLATness:AUTO`

Arguments

None

Examples

`DISP:OFDM:FLAT:AUTO` automatically rescales the horizontal and vertical axes for the best display.

DISPlay:OFDM:FLATness:X:AUTO (No Query Form)

Rescales the horizontal axis automatically in the OFDM Spectral Flatness display.

Conditions

Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group

Display commands

Syntax

`DISPlay:OFDM:FLATness:X:AUTO`

Arguments

None

Examples

`DISP:OFDM:FLAT:X:AUTO` rescales the horizontal axis automatically for the best display.

DISPlay:OFDM:FLATness:X:OFFSet

Sets or queries the value of the offset of the horizontal axis from the center of the OFDM WLAN Spectral Flatness display.

Conditions

Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group

Display commands

Syntax

```
DISPlay:OFDM:FLATness:X:OFFSet <NRf>
DISPlay:OFDM:FLATness:X:OFFSet?
```

Arguments

Floating point number between limits which are set as a function of the current scale value guaranteed to keep the plot at least partially visible on the display. Positive inputs push the plot down (as viewed by the user) and negative inputs push the plot up. 0 recenters the plot.

Returns

The value of the offset of the X axis from the center of the plot as a floating point number.

Examples

`DISP:OFDM:FLAT:X:OFFS 150` shifts the display to the left by 150%.

DISPlay:OFDM:FLATness:X[:SCALe]

Sets or queries the value of the horizontal scale in the OFDM Spectral Flatness display. The units are Frequency or Subcarrier. To set the units, use the command [SENSe]:OFDM:UNIT:FREQuency.

Conditions

Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group

Display commands

Syntax

```
DISPlay:OFDM:FLATness:X[:SCALe] <NRf>
DISPlay:OFDM:FLATness:X[:SCALe]?
```

Arguments

Floating point number that represents the value of the horizontal scale.

Examples

`DISP:OFDM:FLAT:X:SCALE 32` sets the horizontal scale to 32.

DISPlay:OFDM:FLATness:Y:AUTO (No Query Form)

Rescales the vertical axis scale and position values automatically in the OFDM Spectral Flatness display.

Conditions

Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group

Display commands

Syntax

```
DISPlay:OFDM:FLATness:Y:AUTO
```

Arguments

None

Examples

`DISP:OFDM:FLAT:Y:AUTO` automatically selects the vertical scale and position values.

DISPlay:OFDM:FLATness:Y:OFFSet

Sets or queries the value of the offset from the center (vertical position), in percent terms, for the OFDM Spectral Flatness display. To set the Vertical Scale, use the command `DISPlay:OFDM:FLATness:Y[:SCALe]`.

Conditions

Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group

Display commands

Syntax

```
DISPlay:OFDM:FLATness:Y:OFFSet <NRf>
DISPlay:OFDM:FLATness:Y:OFFSet?
```

Arguments

Floating point number between limits which are set as a function of the current scale value guaranteed to keep the plot at least partially visible on the display. Positive inputs push the plot down (as viewed by the user) and negative inputs push the plot up. 0 re-centers the plot.

Returns

The value of the offset from the center of the Y axis for the plot as a floating point number.

Examples

`DISP:OFDM:FLAT:Y:OFFS 150` sets the offset from the center (vertical position) of the plot to 150%.

DISPlay:OFDM:FLATness:Y[:SCALe]

Sets or queries the vertical scale value for the OFDM Spectral Flatness display. To set the vertical position (offset) value, use the command `DISPlay:OFDM:FLATness:Y:OFFSet`.

Conditions

Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group

Display commands

Syntax

```
DISPlay:OFDM:FLATness:Y[:SCALe] <NRf>
DISPlay:OFDM:FLATness:Y[:SCALe]?
```

Arguments

Floating point number that specifies the vertical scale value.

Examples

`DISP:OFDM:FLAT:Y:SCALE 10` sets the vertical scale value to 10.

DISPlay:OFDM:MEASview:DELete (No Query Form)

Deletes the specified OFDM view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:MEASview:DELete { CONSte | SUMMary | STABle | CRESponse | EVM | PERRor | MERRor |
    POWer }
```

Arguments

`CONSte` deletes the OFDM Constellation view.

`SUMMary` deletes the OFDM Summary view.

`STABle` deletes the OFDM Symbol table view.

`CRESponse` deletes the OFDM Channel response view.

`EVM` deletes the EVM (Error Vector Magnitude) versus Time view.

`PERRor` deletes the Phase error versus Time view.

`MERRor` deletes the Magnitude error versus Time view.

`POWer` deletes the OFDM Power view.

If you attempt to delete a view that is not displayed on screen, the error (-200, "Execution error; Measurement not running") will be returned.

Examples

`DISPLAY:OFDM:MEASVIEW:DELETE CONSte` deletes the OFDM Constellation view.

DISPlay:OFDM:MEASview:NEW (No Query Form)

Displays a new OFDM measurement view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:MEASview:NEW { CONSte | SUMMary | STABle | CRESponse | EVM | PERRor | MERRor | POWer }
```

Arguments

`CONSte` creates a new OFDM Constellation view.

`SUMMary` creates a new OFDM Summary view.

`STABle` creates a new OFDM Symbol table view.

`CRESponse` creates a new OFDM Channel response view.

`EVM` creates a new EVM (Error Vector Magnitude) versus Time view.

`PERRor` creates a new Phase error versus Time view.

`MERRor` creates a new Magnitude error versus Time view.

`POWer` creates a new OFDM Power view.

If you attempt to open a view that is currently displayed on screen, the error (–200, "Execution error; Measurement is already running") will be returned.

Examples

`DISPLAY:OFDM:MEASVIEW:NEW:CONSte` creates a new OFDM Constellation view.

DISPlay:OFDM:MEASview:SELect

Selects a OFDM measurement view on the screen. The query returns the currently selected view.

Selecting a measurement optimizes it. Other measurements may be optimized as a side effect. Refer to the `DISPlay:WINDow:OPTimized:MEASurement?` query.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:MEASview:SELect { CONSte | SUMMary | STABle | CRESponse | EVM | PERRor | MERRor |
    POWer }
DISPlay:OFDM:MEASview:SELect?
```

Arguments

`CONSte` creates a new OFDM Constellation view.

`SUMMary` creates a new OFDM Summary view.

`STABle` creates a new OFDM Symbol table view.

`CRESponse` creates a new OFDM Channel response view.

`EVM` creates a new EVM (Error Vector Magnitude) versus Time view.

`PERRor` creates a new Phase error versus Time view.

`MERRor` creates a new Magnitude error versus Time view.

`POWer` creates a new OFDM Power view.

If you attempt to open a view that is currently displayed on screen, the error (-200, "Execution error; Measurement is already running") will be returned.

Examples

`DISPLAY:OFDM:MEASVIEW:SELECT:STABle` selects the OFDM symbol table view.

DISPlay:OFDM:MERRor:FREQuency[:SCALe]

Sets or queries the horizontal range of the OFDM Magnitude error graph.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:MERRor:FREQuency[:SCALe] <value>
DISPlay:OFDM:MERRor:FREQuency[:SCALe]?
```

Arguments

`<value>::= <NRf>` specifies the horizontal range.

Examples

`DISPLAY:OFDM:MERROR:FREQUENCY[:SCALE] 10E+6` sets the horizontal range to 10 MHz.

DISPlay:OFDM:MERRor:FREQuency[:SCALe]:AUTO (No Query Form)

Rescales the frequency automatically to fit the waveform to the screen in the OFDM Magnitude error graph.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:MERRor:FREQuency[:SCALe]:AUTO
```

Arguments

None

Examples

`DISPLAY:OFDM:MERROR:FREQUENCY[:SCALE]:AUTO` rescales the frequency scale automatically to fit the waveform to the screen.

DISPlay:OFDM:MERRor:FREQuency[:SCALe]:OFFSet

Sets or queries the frequency offset in the OFDM Magnitude error graph.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:MERRor:FREQuency[:SCALe]:OFFSet <value>
DISPlay:OFDM:MERRor:FREQuency[:SCALe]:OFFSet?
```

Arguments

`<value>::<NRf>` specifies the frequency offset.

Examples

`DISPLAY:OFDM:MERROR:FREQUENCY[:SCALE]:OFFSET -26` sets the offset to −26 subcarriers.

DISPlay:OFDM:MERRor:MARKer:SHOW:STATe

Determines whether to show or hide the readout for the selected marker in the OFDM Magnitude error view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:MERRor:MARKer:SHOW:STATe { OFF | ON | 1| 0 }
DISPlay:OFDM:MERRor:MARKer:SHOW:STATe?
```

Arguments

`OFF` or 0 hides the readout for the selected marker in the graph.

`ON` or 1 shows the readout for the selected marker in the graph.

Examples

`DISPLAY:OFDM:MERROR:MARKER:SHOW:STATE ON` shows the readout for the selected marker in the graph.

DISPlay:OFDM:MERRor:TIME[:SCALe]

Sets or queries the time scale in the OFDM Magnitude error view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:MERRor:TIME[:SCALe] <value>
DISPlay:OFDM:MERRor:TIME[:SCALe]?
```

Arguments

<value>::= <NRf> specifies the time scale.

Examples

DISPLAY:OFDM:MERROR:TIME[:SCALE] 50 sets the time scale to 50 subcarriers.

DISPlay:OFDM:MERRor:TIME[:SCALe]:AUTO (No Query Form)

Rescales the time automatically to fit the waveform to the screen in the OFDM Magnitude error view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

DISPlay:OFDM:MERRor:TIME[:SCALe]:AUTO

Arguments

None

Examples

DISPLAY:OFDM:MERROR:TIME[:SCALE]:AUTO rescales the time scale automatically to fit the waveform to the screen.

DISPlay:OFDM:MERRor:TIME[:SCALe]:OFFSet

Sets or queries the time offset in the OFDM Magnitude error view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

DISPlay:OFDM:MERRor:TIME[:SCALe]:OFFSet <value>
DISPlay:OFDM:MERRor:TIME[:SCALe]:OFFSet?

Arguments

<value>::= <NRf> specifies the time offset.

Examples

DISPLAY:OFDM:MERROR:TIME[:SCALE]:OFFSET -47 sets the offset to −47 symbols.

DISPlay:OFDM:MERRor:WINDow:SELect:PLOT

Sets or queries the plot in the OFDM Magnitude error window measurement.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:MERRor:WINDow:SELect:PLOT { BOTH | SCARrier | SYMBols }
DISPlay:OFDM:MERRor:WINDow:SELect:PLOT?
```

Arguments

`SCARrier` selects the subcarrier display.

`SYMBols` selects the symbols display

`BOTH` selects both displays.

Examples

`DISPLAY:OFDM:MERROR:WINDOW:SELECT:PLOT SYMBols` selects the symbols display.

DISPlay:OFDM:MERRor:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule in the OFDM Magnitude error view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:MERRor:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 1| 0}
DISPlay:OFDM:MERRor:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or 0 hides the graticule in the graph.

`ON` or 1 shows the graticule in the graph.

Examples

`DISPLAY:OFDM:MERROR:WINDOW:TRACE:GRATICULE:GRID:STATE OFF` turns off the graticule in the graph.

DISPlay:OFDM:MERRor:Y:AUTO (No Query Form)

Rescales the vertical scale automatically to fit the waveform in the OFDM Magnitude error view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:MERRor:Y:AUTO
```

Arguments

None

Examples

`DISPLAY:OFDM:MERROR:Y:AUTO` rescales the vertical scale to fit the waveform on the screen.

DISPlay:OFDM:MERRor:Y:OFFSet

Sets or queries the vertical offset in the OFDM Magnitude error view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:MERRor:Y:OFFSet <value>
DISPlay:OFDM:MERRor:Y:OFFSet?
```

Arguments

`<value>::= <NRf>` specifies the vertical offset.

Examples

`DISPLAY:OFDM:MERROR:Y:OFFSET -90` sets the vertical scale offset to −90%.

DISPlay:OFDM:MERRor:Y[:SCALe]

Sets or queries the vertical range of the OFDM Magnitude error view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:MERRor:Y[:SCALe] <value>
DISPlay:OFDM:MERRor:Y[:SCALe]?
```

Arguments

`<value>::= <NRf>` specifies the vertical scale.

Examples

`DISPLAY:OFDM:MERROR:Y[:SCALE] 100` sets the vertical scale to 100%.

DISPlay:OFDM:PERRor:FREQuency[:SCALe]

Sets or queries the horizontal range of the OFDM Phase error graph.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:PERRor:FREQuency[:SCALe] <value>
DISPlay:OFDM:PERRor:FREQuency[:SCALe]?
```

Arguments

`<value>::= <NRf>` specifies the horizontal range.

Examples

`DISPLAY:OFDM:PERROR:FREQUENCY[:SCALE] 10E+6` sets the horizontal range to 10 MHz.

DISPlay:OFDM:PERRor:FREQuency[:SCALe]:AUTO (No Query Form)

Rescales the frequency automatically to fit the waveform to the screen in the OFDM Phase error graph.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:PERRor:FREQuency[:SCALe]:AUTO
```

Arguments

None

Examples

`DISPLAY:OFDM:PERROR:FREQUENCY[:SCALE]:AUTO` rescales the frequency scale automatically to fit the waveform to the screen.

DISPlay:OFDM:PERRor:FREQuency[:SCALe]:OFFSet

Sets or queries the frequency offset in the OFDM Phase error view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:PERRor:FREQuency[:SCALe]:OFFSet <value>
DISPlay:OFDM:PERRor:FREQuency[:SCALe]:OFFSet?
```

Arguments

`<value>::= <NRf>` specifies the frequency offset.

Examples

`DISPLAY:OFDM:PERROR:FREQUENCY[:SCALE]:OFFSET -26` sets the frequency scale offset to −26 subcarriers.

DISPlay:OFDM:PERRor:MARKer:SHOW:STATe

Determines whether to show or hide the readout for the selected marker in the OFDM Phase error view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:PERRor:MARKer:SHOW:STATe { OFF | ON | 1| 0 }
DISPlay:OFDM:PERRor:MARKer:SHOW:STATe?
```

Arguments

`OFF` or 0 hides the readout for the selected marker in the graph.

`ON` or 1 shows the readout for the selected marker in the graph.

Examples

`DISPLAY:OFDM:PERROR:MARKER:SHOW:STATE ON` shows the readout for the selected marker in the graph.

DISPlay:OFDM:PERRor:TIME[:SCALe]

Sets or queries the time scale in the OFDM Phase error view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:PERRor:TIME[:SCALe] <value>
DISPlay:OFDM:PERRor:TIME[:SCALe]?
```

Arguments

`<value>::= <NRf>` specifies the time scale.

Examples

`DISPLAY:OFDM:PERROR:TIME[:SCALE] 50` sets the time scale to 50 symbols.

DISPlay:OFDM:PERRor:TIME[:SCALe]:AUTO (No Query Form)

Rescales the time automatically to fit the waveform to the screen in the OFDM Phase error view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:PERRor:TIME[:SCALe]:AUTO
```

Arguments

None

Examples

`DISPLAY:OFDM:PERROR:TIME[:SCALE]:AUTO` rescales the time scale automatically to fit the waveform to the screen.

DISPlay:OFDM:PERRor:TIME[:SCALe]:OFFSet

Sets or queries the time offset in the OFDM Phase error view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:PERRor:TIME[:SCALe]:OFFSet <value>
DISPlay:OFDM:PERRor:TIME[:SCALe]:OFFSet?
```

Arguments

`<value>::= <NRf>` specifies the time offset.

Examples

`DISPLAY:OFDM:PERROR:TIME[:SCALE]:OFFSET -17` sets the time scale offset to −17 symbols.

DISPlay:OFDM:PERRor:WINDow:SELect:PLOT

Sets or queries the plot in the OFDM Phase error window measurement.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:PERRor:WINDow:SELect:PLOT { BOTH | SCARrier | SYMBols }
DISPlay:OFDM:PERRor:WINDow:SELect:PLOT?
```

Arguments

`SCARrier` selects the subcarrier display.

`SYMBols` selects the symbols display

`BOTH` selects both displays.

Examples

`DISPLAY:OFDM:PERROR:WINDOW:SELECT:PLOT SYMBols` selects the symbols display.

DISPlay:OFDM:PERRor:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule in the OFDM Phase error view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:PERRor:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 1| 0}
DISPlay:OFDM:PERRor:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or 0 hides the graticule in the graph.

`ON` or 1 shows the graticule in the graph.

Examples

`DISPLAY:OFDM:PERROR:WINDOW:TRACE:GRATICULE:GRID:STATE OFF` turns off the graticule in the graph.

DISPlay:OFDM:PERRor:Y:AUTO (No Query Form)

Rescales the vertical scale automatically to fit the waveform in the OFDM Phase error view.

Conditions

Measurement views: OFDM

Group

Display commands

```
Syntax
DISPlay:OFDM:PERRor:Y:AUTO
```

Arguments

None

Examples

`DISPLAY:OFDM:PERROR:Y:AUTO` rescales the vertical scale to fit the waveform on the screen.

DISPlay:OFDM:PERRor:Y:OFFSet

Sets or queries the vertical offset in the OFDM Phase error view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:PERRor:Y:OFFSet <value>
DISPlay:OFDM:PERRor:Y:OFFSet?
```

Arguments

`<value>::= <NRf>` specifies the horizontal range.

Examples

`DISPLAY:OFDM:PERROR:Y:OFFSET -173` sets the vertical offset to –173 °.

DISPlay:OFDM:PERRor:Y[:SCALe]

Sets or queries the vertical range of the OFDM Phase error view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:PERRor:Y[:SCALe] <value>
DISPlay:OFDM:PERRor:Y[:SCALe]?
```

Arguments

`<value>::= <NRf>` specifies the vertical scale.

Examples

`DISPLAY:OFDM:PERROR:Y[:SCALE] 200` sets the vertical scale to 200 °.

DISPlay:OFDM:POWer:FREQuency[:AUTO](No Query Form)

Rescales the frequency automatically to fit the waveform to the screen in the OFDM Power graph.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:POWer:FREQuency[:AUTO]
```

Arguments

None

Examples

`DISPLAY:OFDM:POWER:FREQUENCY[:AUTO]` rescales the frequency scale automatically to fit the waveform to the screen.

DISPlay:OFDM:POWer:FREQuency[:OFFSet]

Sets or queries the frequency offset in the OFDM Power view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:POWer:FREQuency[:OFFSet] <value>
DISPlay:OFDM:POWer:FREQuency[:OFFSet]?
```

Arguments

`<value>::= <NRf>` specifies the frequency offset.

Examples

`DISPLAY:OFDM:POWER:FREQUENCY[:OFFSET] -26` sets the frequency offset to –26 subcarriers.

DISPlay:OFDM:POWer:FREQuency[:SCALe]

Sets or queries the horizontal range of the OFDM Power graph.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:POWer:FREQuency[:SCALe] <value>
DISPlay:OFDM:POWer:FREQuency[:SCALe]?
```

Arguments

`<value>::= <NRf>` specifies the horizontal range.

Examples

`DISPLAY:OFDM:POWER:FREQUENCY:SCALE 10E+6` sets the horizontal range to 10 MHz.

DISPlay:OFDM:POWer:MARKer:SHOW:STATe

Determines whether to show or hide the readout for the selected marker in the OFDM Power view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:POWer:MARKer:SHOW:STATe { OFF | ON | 1| 0 }
```

Arguments

`OFF` or 0 hides the readout for the selected marker in the graph.

`ON` or 1 shows the readout for the selected marker in the graph.

Examples

`DISPLAY:OFDM:POWER:MARKER:SHOW:STATE ON` shows the readout for the selected marker in the graph.

DISPlay:OFDM:POWer:TIME[:AUTO](No Query Form)

Rescales the time automatically to fit the waveform to the screen in the OFDM Power view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:POWer:TIME[:AUTO]
```

Arguments

None

Examples

`DISPLAY:OFDM:POWER:TIME:AUTO` rescales the time scale automatically to fit the waveform to the screen.

DISPlay:OFDM:POWer:TIME:OFFSet

Sets or queries the time offset in the OFDM Power view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:POWer:TIME:OFFSet <value>
DISPlay:OFDM:POWer:TIME:OFFSet?
```

Arguments

`<value>::= <NR1>` specifies the time offset.

Examples

`DISPLAY:OFDM:POWER:TIME:OFFSET?` might return `2.00000` indicating the offset is 2 symbols.

DISPlay:OFDM:POWer:TIME[:SCALe]

Sets or queries the time scale in the OFDM Phase error view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:POWer:TIME[:SCALe] <value>
DISPlay:OFDM:POWer:TIME[:SCALe]?
```

Arguments

`<value>::= <NRf>` specifies the time scale.

Examples

`DISPLAY:OFDM:POWER:TIME[:SCALE] 46` sets the time scale to 46 symbols.

DISPlay:OFDM:POWer:WINDow:SELect:PLOT

Sets or queries the plot in the OFDM Power window measurement.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:POWer:WINDow:SELect:PLOT { BOTH | SCARrier | SYMBols }
DISPlay:OFDM:POWer:WINDow:SELect:PLOT?
```

Arguments

`SCARrier` selects the subcarrier display.

`SYMBols` selects the symbols display

`BOTH` selects both displays.

Examples

`DISPLAY:OFDM:POWER:WINDOW:SELECT:PLOT SYMBols` selects the symbols display.

DISPlay:OFDM:POWer:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule in the OFDM Power view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:POWer:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 1| 0}
DISPlay:OFDM:POWer:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or 0 hides the graticule in the graph.

`ON` or 1 shows the graticule in the graph.

Examples

`DISPLAY:OFDM:POWER:WINDOW:TRACE:GRATICULE:GRID:STATE OFF` turns off the graticule in the graph.

DISPlay:OFDM:POWer:Y:AUTO (No Query Form)

Rescales the vertical scale automatically to fit the waveform in the OFDM Power view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:POWer:Y:AUTO
```

Arguments

None

Examples

`DISPLAY:OFDM:POWER:Y:AUTO` rescales the vertical scale to fit the waveform on the screen.

DISPlay:OFDM:POWer:Y:OFFSet

Sets or queries the vertical offset in the OFDM Power view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:POWer:Y:OFFSet <value>
DISPlay:OFDM:POWer:Y:OFFSet?
```

Arguments

`<value>::= <NRf>` specifies the horizontal range.

Examples

`DISPLAY:OFDM:POWER:Y:OFFSET -37.29` sets the vertical offset to –37.29 dBm.

DISPlay:OFDM:POWer:Y[:SCALe]

Sets or queries the vertical range of the OFDM Power view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:POWer:Y[:SCALe] <value>
DISPlay:OFDM:POWer:Y[:SCALe]?
```

Arguments

`<value>::= <NRf>` specifies the vertical scale.

Examples

`DISPLAY:OFDM:POWER:Y:SCALE 64.48` sets the vertical scale to 64.48 dB.

DISPlay:P25:CONSte:WINDow:TRACe:GRATicule:GRID:STATe

Sets or queries to show or hide the graticule grid on the screen in the P25 Constellation display.

Conditions

Measurement view: P25 Constellation

Group

Display commands

Syntax

```
DISPlay:P25:CONSte:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:P25:CONSte:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or `0` hides the graticule grid.

`ON` or `1` shows the graticule grid.

Returns

0 means that the graticule grid is hidden.

1 means that the graticule grid is showing.

Examples

`DISPLAY:P25:CONSTE:WINDOW:TRACE:GRATICULE:GRID:STATE ON` shows the graticule grid on the screen.

DISPlay:P25:EDIagram:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen in the P25 Eye Diagram display.

Conditions

Measurement view: P25 Eye Diagram

Group

Display commands

Syntax

```
DISPlay:P25:EDIagram:WINDow:TRACe:GRATicule:GRID:STATe {OFF|ON|0|1}
DISPlay:P25:EDIagram:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or 0 hides the graticule grid.

`ON` or 1 shows the graticule grid.

Returns

0 means that the graticule grid is hidden.

1 means that the graticule grid is showing.

Examples

`DISPlay:P25:EDIagram:WINDow:TRACe:GRATicule:GRID:STATe ON` shows the graticule grid on the screen in the P25 Eye Diagram display.

DISPlay:P25:EDIagram:Y[:SCALe]

Sets or queries the vertical scale (Hz) of the P25 Eye Diagram display.

Conditions

Measurement view: P25 Eye Diagram

Group

Display commands

Syntax

```
DISPlay:P25:EDIagram:Y[:SCALe]<NRf>
DISPlay:P25:EDIagram:Y[:SCALe]?
```

Arguments

`<value>::<NRf>` specifies the vertical scale value (no units).

Range is 1 to 100

Returns

`<NR3>` = a floating point value with an exponent, which is the vertical scale value in Hz.

Examples

`DISP:P25:EDI:Y:SCAL 0.0005` sets the vertical scale to 500 µHz in the display.

`DISP:P25:EDI:Y:SCAL?` might return `500.0000000000E-6`, indicating that the vertical scale is 500 µHz.

DISPlay:P25:EDIagram:Y[:SCALe]:AUTO (No Query Form)

Sets the vertical scale automatically to fit the waveform to the screen in the P25 Eye Diagram display.

Conditions

Measurement view: P25 Eye Diagram

Group

Display commands

Syntax

```
DISPlay:P25:EDIagram:Y[:SCALe]:AUTO
```

Arguments

None.

Returns

0 indicates the vertical scale is set to automatically fit the waveform to the screen in the P25 Eye Diagram display.

1 indicates the vertical scale is not set to automatically fit the waveform to the screen in the P25 Eye Diagram display.

Examples

`DISPLAY:P25:EDIAGRAM:Y[:SCALE]:AUTO` sets the vertical scale automatically to fit the waveform to the screen.

DISPlay:P25:EDIagram:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (center point of the vertical axis) in the P25 Eye Diagram display.

Conditions

Measurement view: P25 Eye Diagram

Group

Display commands

Syntax

```
DISPlay:P25:EDIagram:Y[:SCALe]:OFFSet
DISPlay:P25:EDIagram:Y[:SCALe]:OFFSet?
```

Arguments

`<value>::<NRf>` specifies the minimum vertical value. Range is -50 to +50 (no units).

Returns

Vertical offset value.

Examples

`DISPLAY:P25:EDIAGRAM:Y[:SCALE]:OFFSET -0.5` sets the vertical offset to -0.5 in the P25 Eye Diagram display.

DISPlay:P25:MEASview:DELete (No Query Form)

Deletes the specified P25 measurement display.

Conditions

Measurement view: P25 measurements

Group

Display commands

Syntax

```
DISPlay:P25:MEASview:DELete { FDVT | EDI | PVT | CONS | SUMM }
```

Related Commands

DISPlay:P25:MEASview:NEW (on page 858)

DISPlay:P25:MEASview:SELect (on page 859)

Arguments

`FDVT`: Frequency Deviation vs Time display

`EDI`: P25 Eye Diagram display

`PVT`: Power vs Time display

`CONS`: P25 Constellation display

`SUMM`: P25 Summary display

Examples

`DISPLAY:P25:MEASVIEW:DELETE CONS` deletes the P25 Constellation display.

DISPlay:P25:MEASview:NEW (No Query Form)

Displays a new P25 measurement view.

Conditions

Measurement view: P25 measurements

Group

Display commands

Syntax

```
DISPlay:P25:MEASview:NEW { FDVT | EDI | PVT | CONS | SUMM }
```

Related Commands

DISPlay:P25:MEASview:SELect (on page 859)

DISPlay:P25:MEASview:DELete (on page 857)

Arguments

`FDVT`: Frequency Deviation vs Time display

`EDI`: P25 Eye Diagram display

`PVT`: Power vs Time display

`CONS`: P25 Constellation display

`SUMM`: P25 Summary display

NOTE: If you attempt to open a view that is currently displayed on the screen, the error -200, "Execution error; Measurement is already running" will be returned.

Examples

`DISPLAY:P25:MEASVIEW:NEW PVT` creates a new view of the Power vs Time display.

DISPlay:P25:MEASview:SELect

Selects a P25 measurement display. The query form returns the currently selected display.

Conditions

Measurement view: P25 measurements

Group

Display commands

Syntax

`DISPlay:P25:MEASview:SELect { FDVT | EDI | PVT | CONS | SUMM }`

Related Commands

DISPlay:P25:MEASview:NEW (on page 858)

DISPlay:P25:MEASview:DELete (on page 857)

Arguments

`FDVT`: Frequency Deviation vs Time display

`EDI`: P25 Eye Diagram display

`PVT`: Power vs Time display

`CONS`: P25 Constellation display

`SUMM`: P25 Summary display

NOTE: If you attempt to select a view that is not displayed on the screen, the error -200, "Execution error; Measurement not running" will be returned.

Returns

FDVT: Frequency Deviation vs Time display

EDI: P25 Eye Diagram display

PVT: Power vs Time display

CONS: P25 Constellation display

SUMM: P25 Summary display

Examples

`DISPLAY:P25:MEASVIEW:SELECT CONS` selects the P25 Constellation display.

DISPlay:P25:PVTime:BURSt:X[:SCALe]

Sets or queries the value of the scale (width) value, in seconds, for the P25 Power vs. Time display when using horizontal Full Burst view.

Conditions

Measurement view: P25 Power vs. Time

Group

Display commands

Syntax

```
DISPlay:P25:PVTime:BURSt:X[:SCALe]
DISPlay:P25:PVTime:BURSt:X[:SCALe]?
```

Related Commands

DISPlay:P25:PVTime:BURSt:X[:SCALe]:AUTO (on page 859)

Arguments

Floating point number that represents the value of the scale (width), in seconds, when using horizontal Full Burst view.

Returns

Horizontal scale value.

Examples

`DISPLAY:P25:PVTIME:BURST:X[:SCALE] 5.0E-6` sets the width of the display to 5.000 µS for the horizontal Full Burst view.

DISPlay:P25:PVTime:BURSt:X[:SCALe]:AUTO (No Query Form)

Automatically sets the starting time (position) and scale (width) values for the best display in the horizontal Full Burst view.

Conditions

Measurement view: P25 Power vs. Time

Group

Display commands

Syntax

```
DISPlay:P25:PVTime:BURSt:X[:SCALe]:AUTO
```

Related Commands

DISPlay:P25:PVTime:BURSt:X[:SCALe] (on page 859)

Examples

`DISPLAY:P25:PVTIME:BURST:X:SCALE:AUTO` automatically sets the starting time (position) and scale (width) values for the best display.

DISPlay:P25:PVTime:BURSt:X[:SCALe]:OFFSet

Sets or queries the starting time (position) value, in seconds, for the P25 Power vs. Time display when using the horizontal Full Burst view.

Conditions

Measurement view: P25 Power vs. Time

Group

Display commands

Syntax

```
DISPlay:P25:PVTime:BURSt:X[:SCALe]:OFFSet
DISPlay:P25:PVTime:BURSt:X[:SCALe]:OFFSet?
```

Related Commands

DISPlay:P25:PVTime:BURSt:X[:SCALe]:AUTO (on page 860)

Arguments

Floating point number that represents the starting time (position) value, in seconds, when using horizontal Full Burst view.

Returns

Horizontal offset value.

Examples

`DISPLAY:P25:PVTIME:BURST:X[:SCALE]:OFFSET 5.0E-6` sets the starting time (position) of the graph to 5.000 µS for horizontal Full Burst view.

DISPlay:P25:PVTime:BURSt:X[:SCALe]:RESet (No Query Form)

Resets the starting time (position) and scale (width) values for the horizontal Full Burst view on the P25 Power vs. Time display.

Conditions

Measurement view: P25 Power vs. Time

Group

Display commands

Syntax

```
DISPlay:P25:PVTime:BURSt:X[:SCALe]:RESet
```

Related Commands

DISPlay:P25:PVTime:BURSt:X[:SCALe]:AUTO (on page 860)

Examples

`DISPLAY:P25:PVTIME:BURST:X:SCALE:RESET` resets the starting time (position) and scale (width) values for the horizontal Full Burst view.

DISPlay:P25:PVTime:FALL:X[:SCALe]

Sets or queries the scale (width) value, in seconds, for the P25 Power vs. Time display when using the horizontal Falling Edge view.

Conditions

Measurement view: P25 Power vs. Time

Group

Display commands

Syntax

```
DISPlay:P25:PVTime:FALL:X[:SCALe]<NRf>
DISPlay:P25:PVTime:FALL:X[:SCALe]?
```

Related Commands

Arguments

Floating point number that represents the scale (width), in seconds, when using

the horizontal Falling Edge view.

Returns

Horizontal scale value.

Examples

`DISPLAY:P25:PVTIME:FALL:X:SCALE 5.0E-6` sets the width of the graph to 5.000 µs when using the horizontal Falling Edge view.

DISPlay:P25:PVTime:FALL:X[:SCALe]:AUTO (No Query Form)

Automatically sets the starting time (position) and scale (width) values for the best display in the horizontal Falling Edge view.

Conditions

Measurement view: P25 Power vs. Time

Group

Display commands

Syntax

```
DISPlay:P25:PVTime:FALL:X[:SCALe]:AUTO
```

Related Commands

DISPlay:P25:PVTime:FALL:X[:SCALe]:RESet (on page 863)

Examples

`DISPLAY:P25:PVTIME:FALL:X:SCALE:AUTO` automatically sets the starting time (position) and scale (width) values for the best display.

DISPlay:P25:PVTime:FALL:X[:SCALe]:OFFSet

Sets or queries the value for the starting time (offset) of the graph, in seconds, for the P25 Power vs. Time display when using the horizontal Falling Edge view.

Conditions

Measurement view: P25 Power vs. Time

Group

Display commands

Syntax

```
DISPlay:P25:PVTime:FALL:X[:SCALe]:OFFSet
```

Related Commands

DISPlay:P25:PVTime:RISE:X[:SCALe] (on page 864)

Arguments

Floating point number that represents the value of the starting time (offset) of the graph, in seconds, when using the horizontal Falling Edge view.

Returns

Horizontal offset value.

Examples

`DISPLAY:P25:PVTIME:FALL:X:SCALE:OFFSET 5.0` sets the starting time (position) of the graph to 5.000 µS when using the horizontal Falling Edge view.

DISPlay:P25:PVTime:FALL:X[:SCALe]:RESet (No Query Form)

Resets the starting time (position) and scale (width) values for the horizontal Falling Edge view on the P25 Power vs. Time display.

Conditions

Measurement view: P25 Power vs. Time

Group

Display commands

Syntax

```
DISPlay:P25:PVTime:FALL:X[:SCALe]:RESet
```

Examples

`DISPLAY:P25:PVTIME:FALL:X:SCALE:RESET` resets the starting time (position) and scale (width) values for the horizontal Falling Edge view.

DISPlay:P25:PVTime:MARKer:SHOW:STATe

Shows or hides the marker readout on the P25 Power vs. Time display.

Conditions

Measurement view: P25 Power vs. Time

Group

Display commands

Syntax

```
DISPlay:P25:PVTime:MARKer:SHOW:STATe
```

Arguments

`ON` or 1 specifies to show the marker readout. OFF or 0 specifies to turn the marker readout off.

Returns

`#<num_digit><num_byte><data(1)><data(2)>..<data(n)>`

Where:

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`DISPLAY:P25:PVTIME:MARKER:SHOW:STATE?` might return #43204 (3204-byte data), which represents the vertical values (power) for the P25 Power vs. Time trace.

DISPlay:P25:PVTime:RISE:X[:SCALe]

Sets or queries the scale (width) value, in seconds, for the P25 Power vs. Time display when using the horizontal Rising Edge view.

Conditions

Measurement view: P25 Power vs. Time

Group

Display commands

Syntax

```
DISPlay:P25:PVTime:RISE:X[:SCALe]
DISPlay:P25:PVTime:RISE:X[:SCALe]?
```

Arguments

Floating point number that represents the scale (width) value, in seconds, for the horizontal Rising Edge view.

Returns

Scale value.

Examples

`DISPLAY:P25:PVTIME:RISE:X:SCALE 5.0E-6` sets the width of the graph to 5.000 µS for the horizontal Rising Edge view.

DISPlay:P25:PVTime:RISE:X[:SCALe]:AUTO (No Query Form)

Sets the scale (width) value, in seconds, for the P25 Power vs. Time display to automatic.

Conditions

Measurement view: P25 Power vs. Time

Group

Display commands

Syntax

```
DISPlay:P25:PVTime:RISE:X[:SCALe]:AUTO
```

Related Commands

DISPlay:P25:PVTime:RISE:X[:SCALe]:RESet (on page 865)

Examples

`DISPLAY:P25:PVTIME:RISE:X:SCALE:AUTO` automatically sets the starting time (position) and scale (width) values for the best display.

DISPlay:P25:PVTime:RISE:X[:SCALe]:OFFSet

Sets or queries the starting time value (offset), in seconds, for the P25 Power vs. Time display when using the horizontal Rising Edge view.

Conditions

Measurement view: P25 Power vs. Time

Group

Display commands

Syntax

`DISPlay:P25:PVTime:RISE:X[:SCALe]:OFFSet`

Related Commands

DISPlay:P25:PVTime:RISE:X[:SCALe] (on page 864)

Arguments

Floating point number that represents the value of the starting time (offset) of the graph, in seconds, for the horizontal Rising Edge view.

Returns

Horizontal scale view

Examples

`DISPLAY:P25:PVTIME:RISE:X:SCALE:OFFSET 5.0e-6` sets the starting time (position) of the graph to 5.000 µS for the horizontal Rising Edge view.

DISPlay:P25:PVTime:RISE:X[:SCALe]:RESet (No Query Form)

Resets the starting time (position) and scale (width) values for the horizontal Rising Edge view on the P25 Power vs. Time display.

Conditions

Measurement view: P25 Power vs. Time

Group

Display commands

Syntax

`DISPlay:P25:PVTime:RISE:X[:SCALe]:RESet`

Related Commands

DISPlay:P25:PVTime:RISE:X[:SCALe]:OFFSet (on page 865)

Examples

`DISPLAY:P25:PVTIME:RISE:X:SCALE:RESET` resets the starting time (position) and scale (width) values for the horizontal Rising Edge view.

DISPlay:P25:PVTime:WINDow:SELect:PLOT

Sets or queries which view to use for the P25 Power vs. Time display. Select from Full Burst, Rising Edge or Falling Edge.

Conditions

Measurement view: P25 Power vs. Time

Group

Display commands

Syntax

```
DISPlay:P25:PVTime:WINDow:SELect:PLOT {BURS|RISE|FALL}
DISPlay:P25:PVTime:WINDow:SELect:PLOT?
```

Related Commands

Arguments

- `BURSt` sets the view to Full Burst, which displays the entire packet, with vertical lines indicating the length of the burst.
- `RISE` sets the view to Rising Edge, which zooms the display into the interval around the burst rising edge, with vertical lines indicating the 10% to 90% Power-On Ramp time.
- `FALL` sets the view to Falling Edge, which zooms the display into the interval around the burst Falling Edge, with vertical lines indicating the 90% to 10% Power-Down Ramp time.

Returns

`DISPlay:P25:PVTime:WINDow:SELect:PLOT BURS` sets the view to Full Burst.

Examples

`DISPLAY:P25:PVTIME:WINDOW:SELECT:PLOT?` might return `RISE`, indicating that the view has been set to Rising Edge.

DISPlay:P25:PVTime:WINDow:TRACe:GRATicule:GRID:STATe

Sets or queries the graticule state in the Power vs Time display to on (showing) or off (hidden).

Conditions

Measurement view: P25 Power vs Time

Group

Display commands

Syntax

```
DISPlay:P25:PVTime:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:P25:PVTime:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

OFF or 0 hides the graticule in the Power vs. Time display.

ON or 1 shows the graticule in the Power vs. Time display.

Returns

0 means the graticule in the Noise Temperature display is hidden (off).

1 means the graticule in the Noise Temperature display is showing (on).

Examples

DISPLAY:P25:PVTIME:WINDOW:TRACE:GRATICULE:GRID:STATE ON will turn on the graticule in the Noise Temperature display.

DISPLAY:P25:PVTIME:WINDOW:TRACE:GRATICULE:GRID:STATE? might return 1, indicating that the graticule in the Noise Temperature display is showing (on).

DISPlay:P25:PVTime:Y[:SCALe]

Sets or queries the vertical scale value for the P25 Power vs. Time display.

Conditions

Measurement view: P25 Power vs. Time

Group

Display commands

Syntax

```
DISPlay:P25:PVTime:Y[:SCALe] <NRf>
DISPlay:P25:PVTime:Y[:SCALe]?
```

Related Commands

DISPlay:P25:PVTime:Y[:SCALe]:PDIVision (on page 868)

Arguments

<NRf> = a numeric value for the vertical scale in dB.

Returns

<NR2> = a floating point number without an exponent, which is the vertical scale value.

Examples

DISP:P25:PVT:Y:SCAL 10.5 sets the vertical scale value to 10.50 dB.

DISP:P25:PVT:Y:SCAL? might return 10, indicating that the vertical scale is 10.00 dB.

DISPlay:P25:PVTime:Y[:SCALe]:AUTO (No Query Form)

Automatically selects the vertical scale and position values for the P25 Power vs Time display.

Conditions

Measurement view: P25 Power vs Time

Group

Display commands

Syntax

```
DISPlay:P25:PVTime:Y[:SCALe]:AUTO
```

Related Commands

DISPlay:P25:PVTime:Y[:SCALe] (on page 867)

Examples

`DISP:P25:PVT:Y:SCAL:AUTO` specifies to reset the vertical scale and position (offset) values for the P25 Power vs. Time display.

DISPlay:P25:PVTime:Y[:SCALe]:OFFSet

Sets or queries the value of the vertical offset (top edge of the vertical axis) for the P25 Power vs. Time display.

Conditions

Measurement view: P25 Power vs. Time

Group

Display commands

Syntax

```
DISPlay:P25:PVTime:Y[:SCALe]:OFFSet<NRf>
DISPlay:P25:PVTime:Y[:SCALe]:OFFSet?
```

Related Commands

DISPlay:P25:PVTime:Y[:SCALe] (on page 867)

Arguments

Floating point number between limits that specifies the value of the vertical offset. Positive inputs move the plot down and negative inputs move the plot up (as viewed by the user).

Returns

The value of the vertical offset for the P25 Power vs. Time display as a floating point number.

Examples

`DISPLAY:P25:PVTIME:Y[:SCALE]:OFFSET -10` sets the vertical offset to −10 dBm.

DISPlay:P25:PVTime:Y[:SCALe]:PDIVision

Sets or queries the value of the vertical scale for the P25 Power vs. Time display, in dB/division. This is only a visual control for panning the graph.

Conditions

Measurement view: P25 Power vs. Time

Group

Display commands

Syntax

```
DISPlay:P25:PVTime:Y[:SCALe]:PDIVision<NRf>
DISPlay:P25:PVTime:Y[:SCALe]:PDIVision?
```

Related Commands

DISPlay:P25:PVTime:Y[:SCALe] (on page 867)

Arguments

Floating point value that represents the value of the vertical scale, in dB/division, for the P25 Power vs. Time display.

Examples

`DISPLAY:P25:PVTIME:Y:SCALE:PDIVISION 20` sets the vertical scale to 20.0 dB/division.

DISPlay:P25:RADix

Set or queries the symbol radix for the P25 Symbol Table display.

Conditions

Measurement view: P25 Symbol Table

Group

Display commands

Syntax

```
DISPlay:P25:RADix { QUAT | BIN | MODS }
```

Arguments

`QUATernary`: sets the symbols base to quaternary.

`BINary`: sets the symbols base to binary.

`MODsymbols`: sets the symbols to modular.

Returns

`QUAT`: means that the symbols base is set to Quaternary.

`BINary`: means that the symbols base is set to Binary.

`MODsymbols`: means that the symbols base is set to ModSymbols (modular).

Examples

```
DISPLAY:P25:RADIX
```

DISPlay:PERRor[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe

Sets or queries the graticule grid view state for the Phase error versus Time view.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Magnitude error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Phase error versus Time).

Group

Display commands

Syntax

```
DISPlay:PERRor[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:PERRor[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or 0 hides the graticule grid.

`ON` or 1 shows the graticule grid.

Returns

OFF or 0

ON or 1

Examples

`DISPlay:PERRor:MEASview2:WINDow:TRACe:GRATicule:GRID:STATe ON` shows the graticule grid on the Phase error versus Time Measview 2.

DISPlay:PERRor[:MEASview<y>]:Y[:SCALe]

Sets or queries the vertical range of the Phase error versus Time graph.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Phase error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Phase error versus Time).

Group

Display commands

Syntax

```
DISPlay:PERRor[:MEASview<y>]:Y[:SCALe] <value>
DISPlay:PERRor[:MEASview<y>]:Y[:SCALe]?
```

Related Commands

DISPlay:PERRor:[MEASview1]:Y[:SCALe]:OFFSet (on page 871)

Arguments

`<value>::<NRf>` specifies the vertical range.

Range: 1 to 100%.

Returns

<NRf>

Examples

`DISPlay:PERRor:MEASview1:Y:SCALe` sets the vertical range to 50% in the Phase error versus Time display Measurement view 1.

DISPlay:PERRor[:MEASview<y>]:Y[:SCALe]:AUTO (No query form)

Sets the vertical scale automatically to fit the waveform to the screen in the Phase error versus Time display.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Phase error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Phase error versus Time).

Group

Display commands

Syntax

```
DISPlay:PERRor[:MEASview<y>]:Y[:SCALe]:AUTO
```

Arguments

None

Examples

`DISPlay:PERRor:MEASview2:Y:SCALe:AUTO` sets the vertical scale automatically to fit the waveform to the screen of Phase error versus Time measurement view 2.

DISPlay:PERRor[:MEASview<y>]:Y[:SCALe]:OFFSet

Sets or queries the minimum vertical value (bottom edge) of the Phase error versus Time graph.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Phase error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Phase error versus Time).

Group

Display commands

Syntax

```
DISPlay:PERRor[:MEASview<y>]:Y[:SCALe]:OFFSet <value>
DISPlay:PERRor[:MEASview<y>]:Y[:SCALe]:OFFSet?
```

Related Commands

DISPlay:PERRor[:MEASview<y>]:Y[:SCALe] (on page 870)

Arguments

`<value>::<NRf>` specifies the minimum vertical value.

Range: -360 to +360°.

Returns

<NRf>

Examples

`DISPlay:PERRor:MEASview2:Y:SCALe:OFFSet? -9.5` sets the minimum vertical value to -9.5% in the Phase error versus Time graph Measurement view 2.

DISPlay:PHVTime[:MEASview<y>]:SELected:TIMestamp

Sets or queries the display of the time stamp in the ogram plot.

Conditions

Applicable measurement view must be active.

Group

Display commands

Syntax

`DISPlay:PHVTime[:MEASview<y>]:SELected:TIMestamp {OFF | ON | 0 | 1}`

`DISPlay:PHVTime[:MEASview<y>]:SELected:TIMestamp?`

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional. By default y is 1.

`OFF` or `0` removes or turns off the time stamp.

`ON` or `1` displays or turns on the time stamp.

Returns

`{OFF | ON | 0 | 1}`

Examples

`DISPlay:PHVTime:MEASview1:SELected:TIMestamp?` queries the display status of the time stamp in the ogram plot view 1.

DISPlay:PHVTime[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions

Measurement views: Phase versus Time

Group

Display commands

Syntax

```
DISPlay:PHVTime[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:PHVTime[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

OFF or 0 hides the graticule grid.

ON or 1 shows the graticule grid.

Examples

DISPLAY:PHVTIME:MEASview1:WINDOW:TRACE:GRATICULE:GRID:STATE ON shows the graticule grid on the Frequency versus Time view.

DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]

Sets or queries the horizontal scale (full-scale time) of the Phase versus Time graph. Programming a specified scale sets DISPlay:PHVTime[:MEASview<y>]:X[:SCALe] AUTO:STATe OFF.

Conditions

Measurement views: Phase versus Time

Group

Display commands

Syntax

```
DISPlay:PHVTime[:MEASview<y>]:X[:SCALe] <value>
DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]?
```

Related Commands

DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:AUTO:STAT (on page 875)

DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:OFFSet (on page 877)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<value>::={ <NRf> | MAXimum | MINimum } specifies the horizontal scale in full-scale time. MAXimum and MINimum represent the upper and lower limits of the setting range, respectively.

Use the DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:MAXimum? (on page 877) and DISPlay:PHVTime[:MEASview<y>]:X[: SCALe]:OFFSet:MINimum? (on page 878) queries to get the upper and lower limit values of the setting range.

Examples

DISPLAY:PHVTIME:MEASview1:X:SCALE 1.5ms sets the horizontal scale to 1.5 ms.

DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:AUTO (No Query Form)

Sets the horizontal scale automatically to fit the waveform to the screen in the Phase versus Time view. Executing this command sets DISPlay:PHVTime[:MEASview<y>]:X [SCALe]:AUTO:STATe ON.

Conditions

Measurement views: Phase versus Time

Group

Display commands

Syntax

```
DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:AUTO
```

Related Commands

DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:AUTO:STATe (on page 875)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Examples

`DISPLAY:PHVTIME:MEASview1:X:SCALE:AUTO` sets the horizontal scale automatically to fit the waveform to the screen.

DISPlay:PHVTime[:MEASview?]:Y[:SCALe]:AXIS:ENABle:REFerence {set and query}

Sets or queries the state of Y reference checkbox.

Conditions

Measurement views: Phase VS Time.

Applicable only for MSO/LPD/SX.

Group

Display commands

Syntax

```
DISPlay:PHVTime[:MEASview?]:Y[:SCALe]:AXIS:ENABle:REFerence {0|1|TRUE |FALSE}
DISPlay:PHVTime[:MEASview?]:Y[:SCALe]:AXIS:ENABle:REFerence? (Query)
```

Arguments

`0` is used to change the 'Y Reference' checkbox state from checked to unchecked.

`1` is used to change the 'Y Reference' checkbox state from unchecked to checked.

Example

`DISPlay:PHVTime:MEASview1:Y:SCALe:AXIS:ENABle:REFerence 1` changes the state of 'Y Reference' checkbox to checked.

DISPlay:PHVTime[:MEASview<y>]:READout[:STATe]

Determines whether to show or hide the measurement parameter readouts in the Phase versus Time view.

Conditions

Measurement views: Phase versus Time

Group

Display commands

Syntax

```
DISPlay:PHVTime[:MEASview<y>]:READout[:STATe] { OFF | ON | 0 | 1 }
DISPlay:PHVTime[:MEASview<y>]:READout[:STATe]?
```

Arguments

`OFF` or `0` hides the parameter readouts.

`ON` or `1` shows the parameter readouts.

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1

Examples

`DISPLAY:PHVTIME:MEASVIEW1:READOUT:STATE ON` shows the parameter readouts on the screen.

DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:AUTO:STATe

Determines whether to set the horizontal scale automatically or manually.

Conditions

Measurement views: Phase versus Time

Group

Display commands

Syntax

```
DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:AUTO:STATe { OFF | ON | 0 | 1 }
DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:AUTO:STATe?
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

`OFF` or 0 specifies that the horizontal scale is set manually. To set it, use the DISPlay:PHVTime[:MEASview<y>]:X[:SCALe] (on page 878) and DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:OFFSet (on page 877) commands.

`ON` or 1 specifies that the horizontal scale is set automatically.

Examples

`DISPLAY:PHVTIME:MEASview1:X:SCALE:AUTO:STATE ON` specifies that the horizontal scale is set automatically.

DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:MAXimum? (Query Only)

Queries the upper limit of the horizontal scale setting range.

Conditions

Measurement views: Phase versus Time

Group

Display commands

Syntax

`DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:MAXimum?`

Related Commands

DISPlay:PHVTime[:MEASview<y>]:X[:SCALe] (on page 873)

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

`<NRf>` The upper limit of the horizontal scale setting range.

Examples

`DISPLAY:PHVTIME:MEASview1:X:SCALE:MAXIMUM?` might return `18.135E-3`, indicating that the upper limit of the horizontal scale setting range is 18.135 ms.

DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:MINimum? (Query Only)

Queries the lower limit of the horizontal scale setting range.

Conditions

Measurement views: Phase versus Time

Group

Display commands

Syntax

`DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:MINimum?`

Related Commands

DISPlay:PHVTime[:MEASview<y>]:X[:SCALe] (on page 873)

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

`<NRf>` The lower limit of the horizontal scale setting range.

Examples

`DISPLAY:PHVTIME:MEASview1:X:SCALE:MINIMUM?` might return `10.0E-9`, indicating that the lower limit of the horizontal scale setting range is 10.0 ns.

DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:OFFSet

Sets or queries the minimum horizontal value (left edge) of the Phase versus Time graph. Programming a specified offset sets `DISPlay:PHVTime:MEASview<y>:X[SCALe]:AUTO:STATe OFF`.

Conditions

Measurement views: Phase versus Time

Group

Display commands

Syntax

```
DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:OFFSet
DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:OFFSet?
```

Related Commands

DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:AUTO:STATe (on page 875)

DISPlay:PHVTime[:MEASview<y>]:X[:SCALe] (on page 873)

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

`<value>::={ <NRf> | MAXimum | MINimum }` specifies the horizontal offset. MAXimum and MINimum represent the upper and lower limits of the setting range, respectively.

Use the DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MAXimum? (on page 877) and DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MINimum? (on page 878) queries to get the upper and lower limit values of the setting range.

Examples

`DISPLAY:PHVTIME:MEASview1:X:SCALE:OFFSET 800ns` sets the minimum horizontal value to 800 ns in the Phase versus Time graph.

DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MAXimum? (Query Only)

Queries the upper limit of the horizontal offset setting range.

Conditions

Measurement views: Phase versus Time

Group

Display commands

Syntax

```
DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MAXimum?
```

Related Commands

DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:OFFSet (on page 877)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

<NRf> The upper limit of the horizontal offset setting range.

Examples

DISPLAY:PHVTIME:MEASview1:X:SCALE:OFFSET:MAXIMUM? might return -1.812E-3, indicating that the upper limit of the horizontal offset setting range is -1.812 ms.

DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MINimum? (Query Only)

Queries the lower limit of the horizontal offset setting range.

Conditions

Measurement views: Phase versus Time

Group

Display commands

Syntax

DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:OFFSet:MINimum?

Related Commands

DISPlay:PHVTime[:MEASview<y>]:X[:SCALe]:OFFSet (on page 877)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

<NRf> The lower limit of the horizontal offset setting range.

Examples

DISPLAY:PHVTIME:MEASview1:X:SCALE:OFFSET:MINIMUM? might return -16.28E-3, indicating that the lower limit of the horizontal offset setting range is -16.28 ms.

DISPlay:PHVTime[:MEASview<y>]:Y[:SCALe]

Sets or queries the vertical range of the Phase versus Time graph.

Conditions

Measurement views: Phase versus Time

Group

Display commands

Syntax

DISPlay:PHVTime[:MEASview<y>]:Y[:SCALe] <value>

```
DISPlay:PHVTime[:MEASview<y>]:Y[:SCALe]?
```

Related Commands

DISPlay:PHVTime[:MEASview<y>]:Y[:SCALe]:OFFSet (on page 880)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<value>::<NRf> specifies the vertical range. Range: 1 to 1T°.

Examples

DISPLAY:PHVTIME:MEASview1:Y:SCALE 180 sets the vertical range to 180° in the Phase versus Time graph.

DISPlay:PHVTime[:MEASview<y>]:Y[:SCALe]:AUTO (No Query Form)

Sets the vertical scale automatically to fit the waveform to the screen in the Phase versus Time view.

Conditions

Measurement views: Phase versus Time

Group

Display commands

Syntax

```
DISPlay:PHVTime[:MEASview<y>]:Y[:SCALe]:AUTO
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Examples

DISPLAY:PHVTIME:MEASview1:Y:SCALE:AUTO sets the vertical scale automatically to fit the waveform to the screen.

DISPlay:PHVTime[:MEASview<y>]:Y[:SCALe]:AXIS

Sets or queries the vertical axis representation.

Conditions

Measurement views: Phase versus Time

Group

Display commands

Syntax

```
DISPlay:PHVTime[:MEASview<y>]:Y[:SCALe]:AXIS { MODulopi | CONTinuous }
DISPlay:PHVTime[:MEASview<y>]:Y[:SCALe]:AXIS?
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

MODulopi (modulo π) shows the phase constrained within ±180 ° along the vertical axis.

CONTinuous shows the phase as continuous quantity along the vertical axis.

Examples

DISPLAY:PHVTIME:MEASview1:Y:SCALE:AXIS MODulopi selects modulo π representation for the vertical axis.

DISPlay:PHVTime[:MEASview<y>]:Y[:SCALe]:AXIS:REFerence

Sets or queries which time point in the analysis period to use as the zero-phase-value reference.

Conditions

Measurement views: Phase versus Time

Group

Display commands

Syntax

```
DISPlay:PHVTime[:MEASview<y>]:Y[:SCALe]:AXIS:REFerence <value>
DISPlay:PHVTime[:MEASview<y>]:Y[:SCALe]:AXIS:REFerence?
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<value>::<NRf> specifies the phase reference time.

Examples

DISPLAY:PHVTIME:MEASview1:Y:SCALE:AXIS:REFERENCE 1.5us sets the phase reference time to 1.5 µs.

DISPlay:PHVTime[:MEASview<y>]:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (the value at the center of the vertical axis) in the Phase versus Time graph.

Conditions

Measurement views: Phase versus Time

Group

Display commands

Syntax

```
DISPlay:PHVTime[:MEASview<y>]:Y[:SCALe]:OFFSet <value>
DISPlay:PHVTime[:MEASview<y>]:Y[:SCALe]:OFFSet?
```

Related Commands

DISPlay:PHVTime[:MEASview<y>]:Y[:SCALe] (on page 878)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<value>::<NRf> specifies the vertical offset. Range: -0.5T to +0.5T°.

Examples

DISPLAY:PHVTIME:MEASview1:Y:SCALE:OFFSET −158.5 sets the vertical offset to −158.5° in the Phase versus Time graph.

DISPlay:PHVTime[:MEASview<y>]:Y[:SCALe]:RESCale (No Query Form)

Rescales the vertical axis automatically to fit the Phase versus Time waveform to the screen.

Conditions

Measurement views: Phase versus Time

Group

Display commands

Syntax

DISPlay:PHVTime[:MEASview<y>]:Y[:SCALe]:RESCale

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Examples

DISPLAY:PHVTIME:MEASview1:Y:SCALE:RESCALE rescales the vertical axis automatically to fit the Phase versus Time waveform to the screen.

DISPlay:PNOise:LEGend:STATe

Determines whether to show or hide the trace legend on the display. The legend indicates the trace detection and function on the screen for each displayed trace.

Conditions

Measurement views: Phase versus Time

Group

Display commands

Syntax

DISPlay:PNOise:LEGend:STATe { OFF | ON | 0 | 1 }
DISPlay:PNOise:LEGend:STATe?

Arguments

OFF or 0 hides the trace legend.

ON or 1 shows the trace legend.

Examples

DISPLAY:PNOISE:LEGEND:STATE ON shows the trace legend on the screen.

DISPlay:PNOise:MARKer:SHOW:STATe

Determines whether to show or hide the readout for the selected marker in the phase noise view.

Conditions

Measurement views: Phase noise

Group

Display commands

Syntax

```
DISPlay:PNOise:MARKer:SHOW:STATe { OFF | ON | 0 | 1 }
DISPlay:PNOise:MARKer:SHOW:STATe?
```

Arguments

`OFF` or 0 hides the readout for the selected marker in the graph.

`ON` or 1 shows the readout for the selected marker in the graph.

Examples

`DISPLAY:PNOISE:MARKER:SHOW:STATE ON` shows the readout for the selected marker in the graph.

DISPlay:PNOise:RESet:SCALe (No Query Form)

Resets the horizontal and vertical scale to the default values described below in the phase noise view.

Vertical offset = -50 dBc/Hz, Vertical scale = 100 dB, Horizontal start = 10 Hz, and Horizontal stop = 1 GHz

Conditions

Measurement views: Phase noise

Group

Display commands

Syntax

```
DISPlay:PNOise:RESet:SCALe
```

Arguments

None

Examples

`DISPLAY:PNOISE:RESET:SCALE` resets the horizontal and vertical scale to the default values.

DISPlay:PNOise:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions

Measurement views: Phase noise

Group

Display commands

Syntax

```
DISPlay:PNOise:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:PNOise:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or 0 hides the graticule grid.

`ON` or 1 shows the graticule grid.

Examples

`DISPLAY:PNOISE:WINDOW:TRACE:GRATICULE:GRID:STATE ON` shows the graticule grid on the screen.

DISPlay:PNOise:X[:SCALe]:AUTO (No Query Form)

Rescales the horizontal axis automatically to fit the waveform to the screen in the phase noise view.

Conditions

Measurement views: Phase noise

Group

Display commands

Syntax

```
DISPlay:PNOise:X[:SCALe]:AUTO
```

Arguments

None

Examples

`DISPLAY:PNOISE:X:SCALE:AUTO` rescales the horizontal scale automatically to fit the waveform to the screen.

DISPlay:PNOise:X[:SCALe]:STARt

Sets or queries the start frequency (left edge) of the phase noise graph.

Conditions

Measurement views: Phase noise

Group

Display commands

Syntax

```
DISPlay:PNOise:X[:SCALe]:STARt <value>
DISPlay:PNOise:X[:SCALe]:STARt?
```

Arguments

`<value>::<NRf>` specifies the start frequency.

Range: 10 mHz to 100 MHz.

Note that (start frequency) = 104 × (stop frequency).

Examples

`DISPLAY:PNOISE:X:SCALE:START 10Hz` sets the start frequency to 10 Hz in the phase noise graph.

DISPlay:PNOise:X[:SCALe]:STOP

Sets or queries the stop frequency (right edge) of the phase noise graph.

Conditions

Measurement views: Phase noise

Group

Display commands

Syntax

```
DISPlay:PNOise:X[:SCALe]:STOP <value>
DISPlay:PNOise:X[:SCALe]:STOP?
```

Arguments

`<value>::<NRf>` specifies the stop frequency.

Range: 100 Hz to 1 THz.

Note that (start frequency) = 104 × (stop frequency).

Examples

`DISPLAY:PNOISE:X:SCALE:STOP 2GHz` sets the stop frequency to 2 GHz in the phase noise graph.

DISPlay:PNOise:Y[:SCALe]

Sets or queries the vertical range of the phase noise graph.

Conditions

Measurement views: Phase noise

Group

Display commands

Syntax

```
DISPlay:PNOise:Y[:SCALe] <value>
DISPlay:PNOise:Y[:SCALe]?
```

Arguments

`<value>::<NRf>` specifies the vertical range. Range: 0.1 to 200 dB.

Examples

`DISPLAY:PNOISE:Y:SCALE 100` sets the vertical range to 100 dB for the phase noise graph.

DISPlay:PNOise:Y[:SCALe]:AUTO (No Query Form)

Rescales the vertical axis automatically to fit the waveform to the screen in the phase noise view.

Conditions

Measurement views: Phase noise

Group

Display commands

Syntax

```
DISPlay:PNOise:Y[:SCALe]:AUTO
```

Arguments

None

Examples

`DISPLAY:PNOISE:Y:SCALE:AUTO` rescales the vertical scale automatically to fit the waveform to the screen.

DISPlay:PNOise:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (the value at the top edge of the vertical axis) of the phase noise graph.

Conditions

Measurement views: Phase noise

Group

Display commands

Syntax

```
DISPlay:PNOise:Y[:SCALe]:OFFSet <value>
DISPlay:PNOise:Y[:SCALe]:OFFSet?
```

Arguments

`<value>::<NRf>` specifies the vertical offset. Range: -200 to +20 dBc/Hz.

Examples

`DISPLAY:PNOISE:Y:SCALE:OFFSET -12.5` sets the vertical offset to -12.5 dBc/Hz for the phase noise graph.

DISPlay:PNOise:Y[:SCALe]:PDIVision

Sets or queries the vertical scale (per division) of the phase noise graph.

Conditions

Measurement views: Phase noise

Group

Display commands

Syntax

```
DISPlay:PNOise:Y[:SCALe]:PDIVision <value>
DISPlay:PNOise:Y[:SCALe]:PDIVision?
```

Arguments

`<value>::<NRf>` specifies the vertical scale (per division).

Range: 0.01 to 20 dB/div.

Examples

`DISPLAY:PNOISE:Y:SCALE:PDIVISION 5` sets the vertical scale to 5 dB/div.

DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]:CURRent:BIN

Sets or queries the current bin number where the indicator is positioned from the histogram display.

Conditions

Measurement views: Cumulative Histogram with the specified Measurement View Number

Group

Display commands

Syntax

```
DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]:CURRent:BIN <value>
DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]:CURRent:BIN?
```

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<NR1>` sets the current bin number.

Examples

`DISPLAY:PULSE:CUMULATIVE:HISTOGRAM:CURRENT:BIN 50` sets the current bin number to 50.

DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]:SCALe:AUTO (No Query Form)

Sets the scale auto value in the histogram display.

Conditions

Measurement views: Cumulative Histogram with the specified Measurement View Number

Group

Display commands

Syntax

```
DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]:SCALe:AUTO <value>
```

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<NR1>`

Examples

`DISPLAY:PULSE:CUMULATIVE:HISTOGRAM:MEASview<y>:SCALE:AUTO 1` sets the auto scale value to 1.

DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]:SCALe:RESEt (No Query Form)

Resets the current scale in the histogram display.

Conditions

Measurement views: Cumulative Histogram with the specified Measurement View Number

Group

Display commands

Syntax

`DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>:]SCALe:RESEt <value>`

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<NR1>`

Examples

`DISPlay:PULSe:CUMulative:HISTogram:MEASview<y>:SCALe:RESEt` sets the scale value to 1.

DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]:SCALe:X:AUTO (No Query Form)

Sets the X auto value in the histogram display.

Conditions

Measurement views: Cumulative Histogram with the specified Measurement View Number

Group

Display commands

Syntax

`DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]:SCALe:X:AUTO <value>`

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

The parameter <x> = 0 to 4.

Returns

`<NR1>`

Examples

`DISPLAY:PULSE:CUMULATIVE:HISTOGRAM:MEASview<y>:SCALE:X:AUTO 1` sets the X auto value to 1.

DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]:SCALe:X:NUMBer

Sets or queries the current X scale number from the histogram display.

Conditions

Measurement views: Cumulative Histogram with the specified Measurement View Number

Group

Display commands

Syntax

```
DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]:SCALe:X:NUMBer <value>
DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]:SCALe:X:NUMBer?
```

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

The parameter <x> = 0 to 4.

Returns

`<NR1>` returns the X scale value.

Examples

`DISPlay:PULSe:CUMulative:HISTogram:MEASview<y>:SCALe:X:NUMBer 50` sets the X scale value to 50.

DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]:SCALe:X:OFFSet

Sets or queries the current X offset number from the histogram display.

Conditions

Measurement views: Cumulative Histogram with the specified Measurement View Number

Group

Display commands

Syntax

```
DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]:SCALe:X:OFFSet <value>
DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]:SCALe:X:OFFSet?
```

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

The parameter <x> = 0 to 4.

Returns

`<NR1>` returns the X offset value.

Examples

`DISPlay:PULSe:CUMulative:HISTogram:MEASview<y>:SCALe:X:OFFSET 2` sets the X scale offset value to 2.

DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]:SCALe:X:RSCale (No Query Form)

Sets the current X reset scale in the histogram display.

Conditions

Measurement views: Cumulative Histogram with the specified Measurement View Number

Group

Display commands

Syntax

`DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]:SCALe:X:RSCale <value>`

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

The parameter <x> = 0 to 4.

Returns

`<NR1>`

Examples

`DISPLAY:PULSE:CUMULATIVE:HISTOGRAM:MEASview<y>:SCALE:X:RSCALE 1` sets the X reset scale value to 1.

DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]:SCALe:Y:AUTO (No Query Form)

Sets the current Y auto value in the histogram display.

Conditions

Measurement views: Cumulative Histogram with the specified Measurement View Number

Group

Display commands

Syntax

`DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]:SCALe:Y:AUTO <value>`

Arguments

`<value>` sets the Y auto value.

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<NR1>`

Examples

`DISPlay:PULSe:CUMulative:HISTogram:MEASview<y>:SCALe:Y:AUTO 1` sets the Y auto value to 1.

DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]:SCALe:Y:FULL

Sets or queries the current Y scale full number from the histogram display.

Conditions

Measurement views:

Cumulative Histogram with the specified Measurement View Number

Group

Display commands

Syntax

```
DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]:SCALe:Y:FULL <value>
DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]:SCALe:Y:FULL?
```

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<NR1>` sets the Y scale full value

Examples

`DISPlay:PULSe:CUMulative:HISTogram:MEASview<y>:SCALe:Y:FULL? 10` sets the Y scale value to 10.

DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]:SCALe:Y:OFFSet

Sets the current Y offset numbering the histogram display.

Conditions

Measurement views: Cumulative Histogram with the specified Measurement View Number

Group

Display commands

Syntax

```
DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]:SCALe:Y:OFFSet<value>
DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]:SCALe:Y:OFFSet?
```

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<NR1>` returns the Y scale offset value.

Examples

`DISPlay:PULSe:CUMulative:HISTogram:MEASview<y>:SCALe:Y:OFFSET 2` sets the Y scale offset value to 2.

DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]:SCALe:Y:RSCale (No Query Form)

Sets the current Y reset scale in the histogram display.

Conditions

Measurement views: Cumulative Histogram with the specified Measurement View Number

Group

Display commands

Syntax

`DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]:SCALe:Y:RSCale <value>`

Arguments

`<value>` sets the Y reset scale value.

Returns

`<NR1>` sets the Y reset scale value.

Examples

`DISPlay:PULSe:CUMulative:HISTogram:MEASview<y>:SCALe:Y:RSCALE 1` sets the Y reset scale value to 1.

DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]:SCALe:Y:STOP? (Query Only)

Queries the minimum vertical value (bottom edge) in the cumulative histogram display.

Conditions

Measurement views: Cumulative Histogram with the specified Measurement View Number

Group

Display commands

Syntax

`DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]:SCALe:Y:STOP?`

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<NRf>` is the minimum vertical value (bottom edge).

Examples

`DISPlay:PULSe:CUMulative:HISTogram:MEASview<y>:SCALe:Y:STOP?` might return 0, indicating that the minimum vertical value is 0 in the pulse cumulative histogram display.

DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe

Sets or queries the current graticule selection from the histogram display.

Conditions

Measurement views: Cumulative Histogram with the specified Measurement View Number

Group

Display commands

Syntax

```
DISPlay:PULSe:CUMulative:HISTogram[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe { 0 | 1 }
```

Arguments

`0` sets the display without graticule (box is unchecked)

`1` sets the display with graticule (checkbox is checked)

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

0 or 1, <NRf>

Examples

`DISPLAY:PULSE:CUMULATIVE:HISTOGRAM:MEASview<y>:WINDOW:TRACE:GRATICULE:GRID:STATE 1` sets the display with graticule option checked.

DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:MFReqerror

Sets or queries the Max Frequency error measurement selection from the Cumulative Statistics display.

Conditions

Measurement views: Pulse Cumulative Statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:MFReqerror <value>
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:MFReqerror
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:MFReqerror?
```

Arguments

`0` unchecks the checkbox

`1` checks the checkbox

Returns

0 or 1, <NRf>

Examples

`DISPlay:PULSe:CUMulative:STATistics:MEASview1:MFReqerror 1` enables the Max Frequency Error measurement.

DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:ATX

Sets or queries the Peak Power measurement selection from the Cumulative Statistics display.

Conditions

Measurement views: Pulse Cumulative Statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:ATX <0 | 1>
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:ATX?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

0 unchecks the checkbox

1 checks the checkbox

Returns

0 or 1, <NRf>

Examples

`DISPlay:PULSe:CUMulative:STATistics:MEASview1:ATX 1` enables the Average Transmitted Power measurement.

DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:AVERage

Sets or queries the Average ON Power measurement selection from the Cumulative Statistics display.

Conditions

Measurement views: Pulse Cumulative Statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:AVERage <0 | 1>
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:AVERage?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

0 unchecks the checkbox

1 checks the checkbox

Returns

0 or 1, <NRf>

Examples

DISPlay:PULSe:CUMulative:STATistics:MEASview1:AVERage 1 enables the Average ON Power measurement.

DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:DRODb

Sets or queries the Droop dB measurement selection from the Cumulative Statistics display.

Conditions

Measurement views: Pulse Cumulative Statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:DRODb <0 | 1>
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:DRODb?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

0 unchecks the checkbox

1 checks the checkbox

Returns

0 or 1, <NRf>

Examples

DISPlay:PULSe:CUMulative:STATistics:MEASview1:DRODb 1 enables the Droop dB measurement.

DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:DROop

Sets or queries the Droop measurement selection from the Cumulative Statistics display.

Conditions

Measurement views: Pulse Cumulative Statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:DROop <0 | 1>
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:DROop?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

0 unchecks the checkbox

1 checks the checkbox

Returns

0 or 1, <NRf>

Examples

DISPlay:PULSe:CUMulative:STATistics:MEASview1:DROop 1 enables the Droop measurement.

DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:DUTPct

Sets or queries the Duty Factor (%) measurement selection from the Cumulative Statistics display.

Conditions

Measurement views: Pulse Cumulative Statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:DUTPct <0 | 1>
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:DUTPct?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

0 unchecks the checkbox

1 checks the checkbox

Returns

0 or 1, <NRf>

Examples

DISPlay:PULSe:CUMulative:STATistics:MEASview1:DUTPct 1 enables the Duty Factor(%) measurement.

DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:DUTRatio

Sets or queries the Duty Factor (Ratio) measurement selection from the Cumulative Statistics display.

Conditions

Measurement views: Pulse Cumulative Statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:DUTRatio <0 | 1>
```

```
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:DUTRatio?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

0 unchecks the checkbox

1 checks the checkbox

Returns

0 or 1, <NRf>

Examples

DISPlay:PULSe:CUMulative:STATistics:MEASview1:DUTRatio 1 enables the Duty Factor (Ratio) measurement.

DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:FABS

Sets or queries the Absolute Frequency measurement selection from the Cumulative Statistics display.

Conditions

Measurement views: Pulse Cumulative Statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:FABS <0 | 1>
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:FABS?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

0 unchecks the checkbox

1 checks the checkbox

Returns

0 or 1, <NRf>

Examples

DISPlay:PULSe:CUMulative:STATistics:MEASview1:FABS 1 enables the Absolute Frequency measurement.

DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:FALL

Sets or queries the Fall Time measurement selection from the Cumulative Statistics display.

Conditions

Measurement views: Pulse Cumulative Statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:FALL <0 | 1>
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:FALL?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

0 unchecks the checkbox

1 checks the checkbox

Returns

0 or 1, <NRf>

Examples

DISPlay:PULSe:CUMulative:STATistics:MEASview1:FALL 1 enables the Fall Time measurement.

DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:FDELta

Sets or queries the Delta Frequency measurement selection from the Cumulative Statistics display.

Conditions

Measurement views: Pulse Cumulative Statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:FDELta <0 | 1>
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:FDELta?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

0 unchecks the checkbox

1 checks the checkbox

Returns

0 or 1, <NRf>

Examples

DISPlay:PULSe:CUMulative:STATistics:MEASview1:FDELta 1 enables the Delta Frequency measurement.

DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:FRDeviation

Sets or queries the Frequency Deviation measurement selection from the Cumulative Statistics display.

Conditions

Measurement views: Pulse Cumulative Statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:FRDeviation <0 | 1>
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:FRDeviation?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

0 unchecks the checkbox

1 checks the checkbox

Returns

0 or 1, <NRf>

Examples

DISPlay:PULSe:CUMulative:STATistics:MEASview1:FRDeviation 1 enables the Frequency Deviation measurement.

DISPLAY:PULSe:CUMulative:STATistics[:MEASview<y>]:IRAMplitude

Sets or queries the Response Amplitude measurement selection from the Cumulative Statistics display.

Conditions

Measurement views: Pulse Cumulative Statistics

Group

Display commands

Syntax

```
DISPLAY:PULSe:CUMulative:STATistics[:MEASview<y>]:IRAMplitude <0 | 1>
DISPLAY:PULSe:CUMulative:STATistics[:MEASview<y>]:IRAMplitude?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

0 unchecks the checkbox

1 checks the checkbox

Returns

0 or 1, <NRf>

Examples

DISPLAY:PULSe:CUMulative:STATistics:MEASview1:IRAMplitude 1 enables the Response Amplitude measurement.

DISPLAY:PULSe:CUMulative:STATistics[:MEASview<y>]:IRTime

Sets or queries the Impulse Response Time measurement selection from the Cumulative Statistics display.

Conditions

Measurement views: Pulse Cumulative Statistics

Group

Display commands

Syntax

```
DISPLAY:PULSe:CUMulative:STATistics[:MEASview<y>]:IRTime <0 | 1>
DISPLAY:PULSe:CUMulative:STATistics[:MEASview<y>]:IRTime?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

0 unchecks the checkbox

1 checks the checkbox

Returns

0 or 1, <NRf>

Examples

DISPLAY:PULSe:CUMulative:STATistics:MEASview1:IRTime 1 enables the Impulse Response Time measurement.

DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:MPHerror

Sets or queries the Max phase error measurement selection from the Cumulative Statistics display.

Conditions

Measurement views: Pulse Cumulative Statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:MPHerror <0 | 1>
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:MPHerror?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

0 unchecks the checkbox

1 checks the checkbox

Returns

<Boolean> , 0 or 1

Examples

`DISPlay:PULSe:CUMulative:STATistics:MEASview1:MPHerror 1` enables the Max phase error measurement.

DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:OVEDb

Sets or queries the Overshoot dB measurement selection from the Cumulative Statistics display.

Conditions

Measurement views: Pulse Cumulative Statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:OVEDb <0 | 1>
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:OVEDb?
```

Arguments

`<y>` = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

`0` unchecks the checkbox

`1` checks the checkbox

Returns

0 or 1, <NRf>

Examples

`DISPlay:PULSe:CUMulative:STATistics:MEASview1:OVEDb 1` enables the Overshoot dB measurement.

DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:OVERshoot

Sets or queries the Overshoot measurement selection from the Cumulative Statistics display.

Conditions

Measurement views: Pulse Cumulative Statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:OVERshoot <0 | 1>
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:OVERshoot?
```

Arguments

`<y>` = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

`0` unchecks the checkbox

`1` checks the checkbox

Returns

0 or 1, <NRf>

Examples

`DISPlay:PULSe:CUMulative:STATistics:MEASview1:OVERshoot 1` enables the Overshoot measurement.

DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:PHDeviation

Sets or queries the Phase Deviation measurement selection from the Cumulative Statistics display.

Conditions

Measurement views: Pulse Cumulative Statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:PHDeviation <0 | 1>
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:PHDeviation?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

0 unchecks the checkbox

1 checks the checkbox

Returns

0 or 1, <NRf>

Examples

`DISPlay:PULSe:CUMulative:STATistics:MEASview1:PHDeviation 1` enables the Phase Deviation measurement.

DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:PPFD

Sets or queries the Pulse-Pulse Frequency Difference measurement selection from the Cumulative Statistics display.

Conditions

Measurement views: Pulse Cumulative Statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:PPFD <0 | 1>
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:PPFD?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

0 unchecks the checkbox

1 checks the checkbox

Returns

0 or 1, <NRf>

Examples

DISPlay:PULSe:CUMulative:STATistics:MEASview1:PPFD 1 enables the Pulse-Pulse Frequency Difference measurement.

DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:PPFRequency

Sets or queries the Pulse-pulse carrier frequency measurement selection from the Cumulative Statistics display.

Conditions

Measurement views: Pulse Cumulative Statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:PPFRequency <0 | 1>
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:PPFRequency?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

0 unchecks the checkbox

1 checks the checkbox

Returns

0 or 1, <NRf>

Examples

DISPlay:PULSe:CUMulative:STATistics:MEASview1:PPFRequency 1 enables the Pulse-pulse carrier frequency phase measurement.

DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:PPOWer

Sets or queries the Peak Power measurement selection from the Cumulative Statistics display.

Conditions

Measurement views: Pulse Cumulative Statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:PPOWer <0 | 1>
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:PPOWer?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

0 unchecks the checkbox

1 checks the checkbox

Returns

0 or 1, <NRf>

Examples

DISPlay:PULSe:CUMulative:STATistics:MEASview1:PPOWer 1 enables the Peak Power measurement.

DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:PPPD

Sets or queries the Pulse-Pulse Phase Difference measurement selection from the Cumulative Statistics display.

Conditions

Measurement views: Pulse Cumulative Statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:PPPD <0 | 1>
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:PPPD?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

0 unchecks the checkbox

1 checks the checkbox

Returns

0 or 1, <NRf>

Examples

DISPlay:PULSe:CUMulative:STATistics:MEASview1:PPPD 1 enables the Pulse-Pulse Phase Difference measurement.

DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:PPPHase

Sets or queries the Pulse-pulse carrier phase measurement selection from the Cumulative Statistics display.

Conditions

Measurement views: Pulse Cumulative Statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:PPPHase <0 | 1>
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:PPPHase?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

0 unchecks the checkbox

1 checks the checkbox

Returns

0 or 1, <NRf>

Examples

DISPlay:PULSe:CUMulative:STATistics:MEASview1:PPPHase 1 enables the Pulse-pulse carrier phase measurement.

DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:RINTerval

Sets or queries the Repetition Interval measurement selection from the Cumulative Statistics display.

Conditions

Measurement views: Pulse Cumulative Statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:RINTerval <0 | 1>
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:RINTerval?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

0 unchecks the checkbox

1 checks the checkbox

Returns

0 or 1, <NRf>

Examples

DISPlay:PULSe:CUMulative:STATistics:MEASview1:RINTerval 1 enables the Repetition Interval measurement.

DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:RIPDb

Sets or queries the Ripple dB measurement selection from the Cumulative Statistics display.

Conditions

Measurement views: Pulse Cumulative Statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:RIPDb <0 | 1>
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:RIPDb?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

0 unchecks the checkbox

1 checks the checkbox

Returns

0 or 1, <NRf>

Examples

DISPlay:PULSe:CUMulative:STATistics:MEASview1:RIPDb 1 enables the Ripple dB measurement.

DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:RIPPle

Sets or queries the Ripple measurement selection from the Cumulative Statistics display.

Conditions

Measurement views: Pulse Cumulative Statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:RIPPle <0 | 1>
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:RIPPle?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

0 unchecks the checkbox

1 checks the checkbox

Returns

0 or 1, <NRf>

Examples

DISPlay:PULSe:CUMulative:STATistics:MEASview1:RIPPle 1 enables the Ripple measurement.

DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:RISE

Sets or queries the Rise Time measurement selection from the Cumulative Statistics display.

Conditions

Measurement views: Pulse Cumulative Statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:RISE <0 | 1>
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:RISE?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

0 unchecks the checkbox

1 checks the checkbox

Returns

0 or 1, <NRf>

Examples

DISPlay:PULSe:CUMulative:STATistics:MEASview1:RISE 1 enables the Rise Time measurement.

DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:RMSFreqerror

Sets or queries the RMS Frequency error measurement selection from the Cumulative Statistics display.

Conditions

Measurement views: Pulse Cumulative Statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:RMSFreqerror <0 | 1>
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:RMSFreqerror?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

0 unchecks the checkbox

1 checks the checkbox

Returns

0 or 1, <NRf>

Examples

`DISPlay:PULSe:CUMulative:STATistics:MEASview1:RMSFreqerror 1` enables the RMS Frequency error measurement.

DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:MFReqerror

Sets or queries the Max Frequency error measurement selection from the Cumulative Statistics display.

Conditions

Measurement views: Pulse Cumulative Statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:MFReqerror <0 | 1>
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:MFReqerror?
```

Arguments

`<y>` = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

`0` unchecks the checkbox

`1` checks the checkbox

Returns

<0 | 1>

Examples

`DISPlay:PULSe:CUMulative:STATistics:MEASview1:MFReqerror 1` enables the Max Frequency Error measurement.

DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:RMSPherror

Sets or queries the Max Frequency error measurement selection from the Cumulative Statistics display.

Conditions

Measurement views: Pulse Cumulative Statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:RMSPherror <0 | 1>
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:RMSPherror?
```

Arguments

`<y>` = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

`0` unchecks the checkbox

1 checks the checkbox

Returns

0 or 1, <NRf>

Examples

DISPlay:PULSe:CUMulative:STATistics:MEASview1:RMSPherror 1 enables the Max Frequency Error measurement.

DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:RRATe

Sets or queries the Repetition Rate measurement selection from the Cumulative Statistics display.

Conditions

Measurement views: Pulse Cumulative Statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:RRATe <0 | 1>
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:RRATe?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

0 unchecks the checkbox

1 checks the checkbox

Returns

0 or 1, <NRf>

Examples

DISPlay:PULSe:CUMulative:STATistics:MEASview1:RRATe 1 enables the Repetition Rate measurement.

DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:WIDTh

Sets or queries the Pulse Width measurement selection from the Cumulative Statistics display.

Conditions

Measurement views: Pulse Cumulative Statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:WIDTh <0 | 1>
DISPlay:PULSe:CUMulative:STATistics[:MEASview<y>]:WIDTh?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

0 unchecks the checkbox

1 checks the checkbox

Returns

0 or 1, <NRf>

Examples

DISPlay:PULSe:CUMulative:STATistics:MEASview1:WIDTh 1 enables the Pulse Width measurement.

DISPlay:PULSe:MEASview<y>:DELete (No Query Form)

Deletes the measurement view in the pulsed RF measurements.

Conditions

Measurement views: Pulsed RF measurements

Group

Display commands

Syntax

DISPlay:PULSe:MEASview<y>:DELete { RESult | TRACe | STATistics}

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

RESult deletes the pulse table view.

TRACe deletes the pulse trace view.

STATistics deletes the pulse statistics view.

If you attempt to delete a view that is not displayed on screen, the error (-200, "Execution error; Measurement not running") will be returned.

Examples

DISPlay:PULSe:MEASview1:DELete deletes the pulse trace view.

DISPlay:PULSe:MEASview<y>[:SOURce<x>]:NEW (No Query Form)

Selects a measurement view in the pulsed RF measurements on the screen. The query command returns the currently selected view.

Conditions

Should be connected to MSO/LPD to Set or Query.

Source keyword is optional, if not present Source1 will be default source.

Measurement views: Pulsed RF with specified Measurement view number

Group

Display commands

Syntax

```
DISPlay:PULSe:MEASview<y>[:SOURce<x>]:NEW { RESult | TRACe | STATistics }
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

<x> = 1 to 8, representing Range of Source.

RESult opens the pulse table view.

TRACe opens the pulse trace view.

STATistics opens the pulse statistics view.

If you attempt to open a view that is currently displayed on screen, the error (-200, "Execution error; Measurement is already running") will be returned.

Examples

DISPlay:PULSe:MEASview2:SOURce1:NEW STATistics creates pulse statistics measurement with source 1.

DISPlay:PULSe:MEASview<y>:SELect

Selects a measurement view in the pulsed RF measurements on the screen. The query command returns the currently selected view.

Conditions

Measurement views: Pulsed RF with specified Measurement view number

Group

Display commands

Syntax

```
DISPlay:PULSe:MEASview<y>:SELect { RESult | TRACe | STATistics}
DISPlay:PULSe:MEASview<y>:SELect?
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

RESult selects the pulse table view.

TRACe selects the pulse trace view.

STATistics selects the pulse statistics view.

If you attempt to select a view that is not displayed on screen, the error (-200, "Execution error; Measurement not running") will be returned.

Returns

<NRf>

Examples

DISPlay:PULSe:MEASview2:SELect TRACe selects the pulse trace Measurement view 2.

DISPlay:PULSe:OGRAm[:MEASview<y>]

Sets the chosen display from the Pulse-Ogram display.

Conditions

Measurement views: Pulse-Ogram

Group

Display commands

Syntax

```
DISPlay:PULSe:OGRAm[:MEASview<y>]<arg1>
DISPlay:PULSe:OGRAm[:MEASview<y>]
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

<arg1> is FVT or TVT or BOTH

Where

TVT = Time Vs Time

FVT = Frequency Vs Time

BOTH = Both

Returns

<String> FVT or TVT or BOTH

Examples

DISPlay:PULSe:OGRAm:MEASview1 BOTH sets the display type to both.

DISPlay:PULSe:OGRAm[:MEASview<y>]:FRAME:INFOrmation

Sets or queries the Frame Info parameter is enabled or not from the Pulse-Ogram display.

Conditions

Measurement views: Pulse-Ogram

Group

Display commands

Syntax

```
DISPlay:PULSe:OGRAm[:MEASview<y>]:FRAME:INFOrmation <0 | 1>
DISPlay:PULSe:OGRAm[:MEASview<y>]:FRAME:INFOrmation?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

0 unchecks the checkbox

1 checks the checkbox

Returns

0 or 1, <NRf>

Examples

`DISPlay:PULSe:OGRAm:MEASview1:FRAME:INFOrmation 1` enables the Frame Info parameter.

DISPlay:PULSe:OGRAm[:MEASview<y>]:MARKer:FRAME

Sets or queries the Marker frame number from the Pulse-Ogram display.

Conditions

Measurement views: Pulse-Ogram

Group

Display commands

Syntax

```
DISPlay:PULSe:OGRAm[:MEASview<y>]:MARKer:FRAME <value>
DISPlay:PULSe:OGRAm[:MEASview<y>]:MARKer:FRAME?
```

Arguments

`<y>` = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

`<value> :: = <NR1>` sets the marker frame number.

Returns

`<NR1>`

Examples

`DISPlay:PULSe:OGRAm:MEASview1:MARKer:FRAME 5` sets the marker frame number to 5.

DISPlay:PULSe:OGRAm[:MEASview<y>]:MARKer:SHOW:STATe

Sets or queries the Show Marker readout in graph option from the Prefs tab of the Pulse-Ogram display.

Conditions

Measurement views: Pulse-Ogram

Group

Display commands

Syntax

```
DISPlay:PULSe:OGRAm[:MEASview<y>]:MARKer:SHOW:STATe {OFF | ON | 0 | 1}
DISPlay:PULSe:OGRAm[:MEASview<y>]:MARKer:SHOW:STATe?
```

Arguments

`<y>` = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

`0` unchecks the checkbox

1 checks the checkbox

Returns

{OFF | ON | 0 | 1}

Examples

DISPlay:PULSe:OGRAm:MEASview1:MARKer:SHOW:STATe 1 enables the Show Marker readout in graph parameter.

DISPlay:PULSe:OGRAm[:MEASview<y>][:SCALe]:X:FVTime:FREQuency:STARt

Sets or queries the current X scale frequency start value from the Pulse-Ogram display.

Conditions

Measurement views: Pulse-Ogram

Group

Display commands

Syntax

```
DISPlay:PULSe:OGRAm[:MEASview<y>][:SCALe]:X:FVTime:FREQuency:STARt <value>
DISPlay:PULSe:OGRAm[:MEASview<y>][:SCALe]:X:FVTime:FREQuency:STARt?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

<value> :: = <NR1> sets the X scale frequency start value.

Returns

<NR1>

Examples

DISPlay:PULSe:OGRAm:MEASview1:SCALe:X:FVTime:FREQuency:STARt 43 sets X scale frequency start value to 43.

DISPlay:PULSe:OGRAm[:MEASview<y>][:SCALe]:X:FVTime:FREQuency:STOP

Sets or queries the current X scale frequency stop value from the Pulse-Ogram display.

Conditions

Measurement views: Pulse-Ogram

Group

Display commands

Syntax

```
DISPlay:PULSe:OGRAm[:MEASview<y>][:SCALe]:X:FVTime:FREQuency:STOP <value>
DISPlay:PULSe:OGRAm[:MEASview<y>][:SCALe]:X:FVTime:FREQuency:STOP?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

<value> :: = <NR1> sets the X scale frequency stop value.

Returns

<NR1>

Examples

DISPlay:PULSe:OGRAm:MEASview1:SCALe:X:FVTime:FREQuency:STOP 155 sets X scale frequency stop value to 155.

DISPlay:PULSe:OGRAm[:MEASview<y>][:SCALe]:X:FVTime:RSCale (No Query Form)

Restores the horizontal axis of start frequency and stop frequency to default value in the Frequency vs Time plot of Pulse-Ogram display.

Conditions

Measurement views: Pulse-Ogram

Group

Display commands

Syntax

DISPlay:PULSe:OGRAm[:MEASview<y>][:SCALe]:X:FVTime:RSCale

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Examples

DISPlay:PULSe:OGRAm:MEASview1:SCALe:X:FVTime:RSCale restores the horizontal axis of start frequency and stop frequency to default value.

DISPlay:PULSe:OGRAm[:MEASview<y>][:SCALe]:X:RSCale (No Query Form)

Restores the horizontal axis of position and scale to default value in the Time vs Time plot of Pulse-Ogram display.

Conditions

Measurement views: Pulse-Ogram

Group

Display commands

Syntax

DISPlay:PULSe:OGRAm[:MEASview<y>][:SCALe]:X:RSCale

Arguments

None

Examples

`DISPLay:PULSe:OGRAm:MEASview1[:SCALe]:X:RSCALe` restores the horizontal axis of position and scale to default value.

DISPlay:PULSe:OGRAm[:MEASview<y>][:SCALe]:X:TVTime (No Query Form)

Sets the scale value in the Time vs Time of Pulse-Ogram display.

Conditions

Measurement views: Pulse-Ogram

Group

Display commands

Syntax

`DISPlay:PULSe:OGRAm[:MEASview<y>][:SCALe]:X:TVTime <value>`

Arguments

`<y>` = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

`<value> :: = <NR1>` sets the scale value in Time vs Time.

Examples

`DISPlay:PULSe:OGRAm:MEASview1:SCALe:X:TVTime 1` sets the scale value to 1.

DISPlay:PULSe:OGRAm[:MEASview<y>][:SCALe]:X:TVTime:OFFSet (No Query Form)

Sets the scale value in the Pulse-Ogram display.

Conditions

Measurement views: Pulse-Ogram

Group

Display commands

Syntax

`DISPlay:PULSe:OGRAm[:MEASview<y>][:SCALe]:X:TVTime:OFFSet <value>`

Arguments

`<y>` = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

`<value> :: = <NR1>` sets the X scale offset value.

Examples

`DISPlay:PULSe:OGRAm:MEASview1:SCALe:X:TVTime:OFFSet 1` sets the X scale offset value to 1.

DISPlay:PULSe:OGRAm[:MEASview<y>][:SCALe]:X:TVTime:RSCale (No Query Form)

Restores the horizontal axis of position and scale to default value in the Time vs Time plot of Pulse-Ogram display.

Conditions

Measurement views: Pulse-Ogram

Group

Display commands

Syntax

```
DISPlay:PULSe:OGRAm[:MEASview<y>][:SCALe]:X:TVTime:RSCale
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Returns

None

Examples

`DISPlay:PULSe:OGRAm:MEASview1:SCALe:X:TVTime:RSCale` restores the horizontal axis of position and scale to default value.

DISPlay:PULSe:OGRAm[:MEASview<y>]:FVTime:DETection

Sets or queries the type of detection for the specified trace.

Conditions

Measurement views: Pulse-Ogram

Group

Display commands

Syntax

```
DISPlay:PULSe:OGRAm[:MEASview<y>]:FVTime:DETection {AVERage | POSitive | NEGative | POSNegative |
   SAMPle}
DISPlay:PULSe:OGRAm[:MEASview<y>]:FVTime:DETection?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

`AVERage` displays the average data value for each pixel.

`POSitive` displays the maximum data value for each pixel.

`NEGative` displays the minimum data value for each pixel.

`POSNegative` displays the maximum and minimum data values for each pixel.

SAMPle displays the first sample value received for each pixel.

Returns

{AVERage | POSitive | NEGative | POSNegative | SAMPle}

Examples

DISPlay:PULSe:OGRAm:MEASview1:FVTime:DETection POSitive enables positive detection.

DISPlay:PULSe:OGRAm[:MEASview<y>][:SCALe]:RSCale

Restores the default value of both the axis inTime vs Time and Frequency vs Time plot of Pulse-Ogram display.

Conditions

Measurement views: Pulse-Ogram

Group

Display commands

Syntax

```
DISPlay:PULSe:OGRAm[:MEASview<y>][:SCALe]:RSCale
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Examples

DISPlay:PULSe:OGRAm:MEASview1:SCALe:RSCale restores default value in PulseOgram displays.

DISPlay:PULSe:OGRAm[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions

Measurement views: Pulse-Ogram

Group

Display commands

Syntax

```
DISPlay:PULSe:OGRAm[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:PULSe:OGRAm[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

OFF or 0 hides the graticule grid.

ON or 1 shows the graticule grid.

Returns

{ OFF | ON | 0 | 1 }

Examples

`DISPlay:PULSe:OGRAm:MEASview1:WINDow:TRACe:GRATicule:GRID:STATe ON` shows the graticule grid on the screen.

DISPlay:PULSe:OGRAm[:MEASview<y>]:TRACE:POINts

Sets or queries the trace points in the Pulseo-gram measurement.

Conditions

Measurement views: Pulse-Ogram

Group

Display commands

Syntax

```
DISPlay:PULSe:OGRAm[:MEASview<y>]:TRACE:POINts {ONEK | TENK | HUNDredk | NDECimate | NEVerdecimate}
DISPlay:PULSe:OGRAm[:MEASview<y>]:TRACE:POINts?
```

Arguments

`<y>` = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

`ONEK` sets the trace points to 1k.

`TENK` sets the trace points to 10k.

`HUNDredk` sets the trace points to 100k.

`NDECimate` or `NEVerdecimate` sets the trace points to Never decimate.

Returns

{ONEK | TENK | HUNDredk | NDECimate | NEVerdecimate}

Examples

`DISPlay:PULSe:OGRAm:MEASview1:TRACE:POINts ONEK` selects the trace points of 1000 in the PulseOgram measurement.

DISPlay:PULSe:OGRAm[:MEASview<y>][:SCALe]:Y:FRAMe:FULL

Sets or queries the frame scale value in the Pulse-Ogram display in fast frame mode acquisition.

Conditions

Measurement views: Pulse-Ogram

Group

Display commands

Syntax

```
DISPlay:PULSe:OGRAm[:MEASview<y>][:SCALe]:Y:FRAMe:FULL <value>
DISPlay:PULSe:OGRAm[:MEASview<y>][:SCALe]:Y:FRAMe:FULL?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

<value> :: = <NR1> sets the frame scale value.

Returns

<NR1> sets the frame scale value.

Examples

DISPlay:PULSe:OGRAm:MEASview1:SCALe:Y:FRAMe:FULL 4 sets the frame scale value to 4.

DISPlay:PULSe:OGRAm:SCALe:Y:FRAMe:OFFSet

Sets or queries the frame offset value in the Pulse-Ogram display in fast frame mode acquisition.

Conditions

Measurement views: Pulse-Ogram

Group

Display commands

Syntax

```
DISPlay:PULSe:OGRAm:SCALe:Y:FRAMe:OFFSet <value>
DISPlay:PULSe:OGRAm:SCALe:Y:FRAMe:OFFSetDISPlay:PULSe:OGRAm: SCALe:Y:FRAMe:OFFSet?
```

Arguments

<value> :: = <NR1> sets the frame offset value.

Examples

DISPLAY:PULSE:OGRAM:SCALE:Y:FRAME:OFFSET 2 sets the frame offset value to 2.

DISPlay:PULSe:OGRAm[:MEASview<y>]:TVTime:SMOOth:POINts

Sets or queries the value of Smooth points from the Traces tab in Time vs Time display of the Pulse-Ogram.

Conditions

Measurement views: Pulse-Ogram

Group

Display commands

Syntax

```
DISPlay:PULSe:OGRAm[:MEASview<y>]:TVTime:SMOOth:POINts <value>
DISPlay:PULSe:OGRAm[:MEASview<y>]:TVTime:SMOOth:POINts?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

<value> :: = <NR1> sets the smooth points value.

Returns

<NR1> sets the smooth points value.

Examples

`DISPlay:PULSe:OGRAm:MEASview1:TVTime:SMOOth:POINts 5` sets the smooth points value to 5.

DISPlay:PULSe:OGRAm[:MEASview<y>]

Sets the chosen display from the Pulse-Ogram display.

Conditions

Measurement views: Pulse-Ogram

Group

Display commands

Syntax

```
DISPlay:PULSe:OGRAm[:MEASview<y>] <arg1>
DISPlay:PULSe:OGRAm[:MEASview<y>]?
```

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

`<arg1>` = `{TVT | FVT | BOTH}`

Where

`TVT` =Time Vs Time

`FVT` =Frequency Vs Time

`BOTH` =Both

Returns

<String> TVT or FVT or BOTH

Examples

`DISPlay:PULSe:OGRAm:MEASview1 FVT` sets Frequency vs Time as the display type.

DISPlay:PULSe:RESult[:MEASview<y>]:ATX

Determines whether or not to show the average transmitted power measurement result in the pulse table.

Conditions

Measurement views: Pulse table

Group

Display commands

Syntax

```
DISPlay:PULSe:RESult[:MEASview<y>]:ATX { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult[:MEASview<y>]:ATX?
```

Arguments

`OFF` or `0` does not show the average transmitted power measurement result.

`ON` or `1` shows the average transmitted power measurement result in the pulse table.

Returns

See Arguments.

Examples

`DISPLAY:PULSE:RESULT:MEASVIEW1:ATX ON` shows the average transmitted power measurement result in the pulse table.

DISPlay:PULSe:RESult[:MEASview<y>]:AVERage

Determines whether or not to show the average on power measurement result in the pulse table.

Conditions

Measurement views: Pulse table

Group

Display commands

Syntax

```
DISPlay:PULSe:RESult[:MEASview<y>]:AVERage { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult[:MEASview<y>]:AVERage?
```

Arguments

`OFF` or `0` does not show the average on power measurement result.

`ON` or `1` shows the average on power measurement result in the results table.

Returns

See Arguments.

Examples

`DISPLAY:PULSE:RESULT:MEASVIEW1:AVERAGE ON` shows the average on power measurement result in the pulse table.

DISPlay:PULSe:RESult[:MEASview<y>]:DRODb

Determines whether or not to show the droop measurement result in dB in the pulse table.

Conditions

Measurement views: Pulse table

Group

Display commands

Syntax

```
DISPlay:PULSe:RESult[:MEASview<y>]:DRODb { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult[:MEASview<y>]:DRODb?
```

Arguments

`OFF` or `0` does not show the droop measurement result in dB.

`ON` or `1` shows the droop measurement result in dB in the pulse table.

Returns

See Arguments.

Examples

`DISPLAY:PULSE:RESULT:MEASVIEW1:DRODB ON` shows the droop measurement result in dB in the pulse table.

DISPlay:PULSe:RESult[:MEASview<y>]:DROop

Sets or queries showing the droop measurement result in the pulse table.

Conditions

Measurement views: Pulse table

Group

Display commands

Syntax

```
DISPlay:PULSe:RESult[:MEASview<y>]:DROop { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult[:MEASview<y>]:DROop?
```

Arguments

`OFF` or `0` does not show the droop measurement result.

`ON` or `1` shows the droop measurement result in the pulse table.

Returns

See Arguments.

Examples

`DISPLAY:PULSE:RESULT:MEASVIEW1:DROOP ON` shows the droop measurement result in the pulse table.

DISPlay:PULSe:RESult[:MEASview<y>]:DUTPct

Determines whether or not to show the duty factor (%) measurement result in the pulse table.

Conditions

Measurement views: Pulse table

Group

Display commands

Syntax

```
DISPlay:PULSe:RESult[:MEASview<y>]:DUTPct { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult[:MEASview<y>]:DUTPct?
```

Arguments

OFF or 0 does not show the duty factor measurement result.

ON or 1 shows the duty factor measurement result in the pulse table.

Returns

See Arguments.

Examples

DISPLAY:PULSE:RESULT:MEASVIEW1:DUTPCT ON shows the duty factor (%) measurement result in the pulse table.

DISPlay:PULSe:RESult[:MEASview<y>]:DUTRatio

Determines whether or not to show the duty factor (ratio) measurement result in the pulse table.

Conditions

Measurement views: Pulse table

Group

Display commands

Syntax

```
DISPlay:PULSe:RESult[:MEASview<y>]:DUTRatio { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult[:MEASview<y>]:DUTRatio?
```

Arguments

OFF or 0 does not show the duty factor measurement result.

ON or 1 shows the duty factor measurement result in the pulse table.

Returns

See Arguments.

Examples

DISPLAY:PULSE:RESULT:MEASVIEW1:DUTRATIO ON shows the duty factor (ratio) measurement result in the pulse table.

DISPlay:PULSe:RESult[:MEASview<y>]:FABS

Determines whether or not to show the absolute frequency measurement result in the Pulse Table.

Conditions

Measurement views: Pulse Table

Group

Display commands

Syntax

```
DISPlay:PULSe:RESult[:MEASview<y>]:FABS {OFF | ON | 0 | 1}
```

Arguments

OFF or 0 does not show the absolute frequency measurement result in the pulse table.

ON or 1 shows the absolute frequency measurement result in the pulse table.

Returns

See Arguments.

Examples

DISPLAY:PULSE:RESULT:MEASVIEW1:FABS ON shows the absolute frequency measurement result in the pulse table.

DISPlay:PULSe:RESult[:MEASview<y>]:FALL

Determines whether or not to show the fall time measurement result in the pulse table.

Conditions

Measurement views: Pulse table

Group

Display commands

Syntax

```
DISPlay:PULSe:RESult[:MEASview<y>]:FALL { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult[:MEASview<y>]:FALL?
```

Arguments

OFF or 0 does not show the fall time measurement result.

ON or 1 shows the fall time measurement result in the pulse table.

Returns

See Arguments.

Examples

DISPLAY:PULSE:RESULT:MEASVIEW1:FALL ON shows the fall time measurement result in the pulse table.

DISPlay:PULSe:RESult[:MEASview<y>]:FDELta

Determines whether or not to show the delta frequency measurement result in the pulse table.

Conditions

Measurement views: Pulse table

Group

Display commands

Syntax

```
DISPlay:PULSe:RESult[:MEASview<y>]:FDELta { OFF | ON | 0 | 1 }
```

Arguments

OFF or 0 does not show the delta frequency measurement result.

ON or 1 shows the delta frequency measurement result in the pulse table.

Returns

See Arguments.

Examples

`DISPLAY:PULSE:RESULT:MEASVIEW1:FDELTA ON` shows the delta frequency measurement result in the pulse table.

DISPlay:PULSe:RESult[:MEASview<y>]:FRDeviation

Determines whether or not to show the frequency deviation measurement result in the pulse table.

Conditions

Measurement views: Pulse table

Group

Display commands

Syntax

```
DISPlay:PULSe:RESult[:MEASview<y>]:FRDeviation { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult[:MEASview<y>]:FRDeviation?
```

Arguments

`OFF` or `0` does not show the frequency deviation measurement result.

`ON` or `1` shows the frequency deviation measurement result in the pulse table.

Returns

See Arguments.

Examples

`DISPLAY:PULSE:RESULT:MEASVIEW1:FRDEVIATION ON` shows the frequency deviation measurement result in the pulse table.

DISPlay:PULSe:RESult[:MEASview<y>]:IRAMplitude

Sets or queries showing the impulse response amplitude measurement result in the pulse table.

Conditions

Measurement views: Pulse table

Group

Display commands

Syntax

```
DISPlay:PULSe:RESult[:MEASview<y>]:IRAMplitude { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult[:MEASview<y>]:IRAMplitude?
```

Arguments

`OFF` or `0` does not show the Impulse Response Amplitude measurement result.

`ON` or `1` shows the Impulse Response Amplitude measurement result in the results table.

Returns

See Arguments.

Examples

`DISPLAY:PULSE:RESULT:MEASVIEW1:IRAMPLITUDE ON` shows the Impulse Response Amplitude measurement result in the pulse table.

DISPlay:PULSe:RESult[:MEASview<y>]:IRTime

Sets or queries showing the Impulse Response Time measurement result in the pulse table.

Conditions

Measurement views: Pulse table

Group

Display commands

Syntax

```
DISPlay:PULSe:RESult[:MEASview<y>]:IRTime { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult[:MEASview<y>]:IRTime?
```

Arguments

`OFF` or `0` does not show the Impulse Response Time measurement result.

`ON` or `1` shows the Impulse Response Time measurement result in the results table.

Returns

See Arguments.

Examples

`DISPLAY:PULSE:RESULT:MEASVIEW1:IRTIME ON` shows the Impulse Response Time measurement result in the pulse table.

DISPlay:PULSe:RESult[:MEASview<y>]:MFReqerror

Determines whether or not to show the maximum frequency error measurement result in the pulse table.

Conditions

Measurement views: Pulse table

Group

Display commands

Syntax

```
DISPlay:PULSe:RESult[:MEASview<y>]:MFReqerror { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult[:MEASview<y>]:MFReqerror?
```

Arguments

`OFF` or `0` does not show the maximum frequency error measurement result.

`ON` or `1` shows the maximum frequency error measurement result in the pulse table.

Returns

See Arguments.

Examples

`DISPLAY:PULSE:RESULT:MEASVIEW1:MFREQERROR ON` shows the maximum frequency error measurement result in the pulse table.

DISPlay:PULSe:RESult[:MEASview<y>]:MPHerror

Determines whether or not to show the maximum phase error measurement result in the pulse table.

Conditions

Measurement views: Pulse table

Group

Display commands

Syntax

```
DISPlay:PULSe:RESult[:MEASview<y>]:MPHerror { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult[:MEASview<y>]:MPHerror?
```

Arguments

`OFF` or `0` does not show the maximum phase error measurement result.

`ON` or `1` shows the maximum phase error measurement result in the pulse table.

Returns

See Arguments.

Examples

`DISPLAY:PULSE:RESULT:MEASVIEW1:MPHERROR ON` shows the maximum phase error measurement result in the pulse table.

DISPlay:PULSe:RESult[:MEASview<y>]:OVEDb

Determines whether or not to show the Overshoot measurement result in dB in the pulse table.

Conditions

Measurement views: Pulse table

Group

Display commands

Syntax

```
DISPlay:PULSe:RESult[:MEASview<y>]:OVEDb { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult[:MEASview<y>]:OVEDb?
```

Arguments

`OFF` or `0` does not show the Overshoot measurement result in dB.

`ON` or `1` shows the Overshoot measurement result in dB in the pulse table.

Returns

See Arguments.

Examples

`DISPLAY:PULSE:RESULT:MEASVIEW1:OVEDB ON` shows the Overshoot measurement result in dB in the pulse table.

DISPlay:PULSe:RESult[:MEASview<y>]:OVERshoot

Sets or queries whether or not to show the Overshoot measurement result in the pulse table.

Conditions

Measurement views: Pulse table

Group

Display commands

Syntax

```
DISPlay:PULSe:RESult[:MEASview<y>]:OVERshoot { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult[:MEASview<y>]:OVERshoot?
```

Arguments

`OFF` or `0` does not show the Overshoot measurement result.

`ON` or `1` shows the Overshoot measurement result in the pulse table.

Returns

See Arguments.

Examples

`DISPLAY:PULSE:RESULT:MEASVIEW1:OVERSHOOT ON` shows the Overshoot measurement result in the pulse table.

DISPlay:PULSe:RESult[:MEASview<y>]:PHDeviation

Determines whether or not to show the phase deviation measurement result in the pulse table.

Conditions

Measurement views: Pulse table

Group

Display commands

Syntax

```
DISPlay:PULSe:RESult[:MEASview<y>]:PHDeviation { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult[:MEASview<y>]:PHDeviation?
```

Arguments

`OFF` or `0` does not show the phase deviation measurement result.

`ON` or `1` shows the phase deviation measurement result in the pulse table.

Returns

See Arguments.

Examples

`DISPLAY:PULSE:RESULT:MEASVIEW1:PHDEVIATION ON` shows the phase deviation measurement result in the pulse table.

DISPlay:PULSe:RESult[:MEASview<y>]:PPPD

Determines whether or not to show the pulse to pulse phase measurement result in the Pulse Table.

Conditions

Measurement views: Pulse Table

Group

Display commands

Syntax

`DISPlay:PULSe:RESult[:MEASview<y>]:PPPD {OFF | ON | 0 | 1}`

Arguments

`OFF` or `0` does not show the pulse to pulse phase measurement result in the pulse table.

`ON` or `1` shows the pulse to pulse phase measurement result in the pulse table.

Returns

See Arguments.

Examples

`DISPLAY:PULSE:RESULT:MEASVIEW1:PPPD ON` shows the pulse to pulse frequency measurement result in the pulse table.

DISPlay:PULSe:RESult[:MEASview<y>]:PPFD

Determines whether or not to show the pulse to pulse frequency measurement result in the Pulse Table.

Conditions

Measurement views: Pulse Table

Group

Display commands

Syntax

`DISPlay:PULSe:RESult[:MEASview<y>]:PPFD {OFF | ON | 0 | 1}`

Arguments

`OFF` or `0` does not show the pulse to pulse frequency measurement result in the pulse table.

`ON` or `1` shows the pulse to pulse frequency measurement result in the pulse table.

Returns

See Arguments.

Examples

`DISPLAY:PULSE:RESULT:MEASVIEW1:PPFD ON` shows the pulse to pulse frequency measurement result in the pulse table.

DISPlay:PULSe:RESult[:MEASview<y>]:PPFRequency

Determines whether or not to show the pulse-pulse carrier frequency measurement result in the pulse table.

Conditions

Measurement views: Pulse table

Group

Display commands

Syntax

```
DISPlay:PULSe:RESult[:MEASview<y>]:PPFRequency { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult[:MEASview<y>]:PPFRequency?
```

Arguments

`OFF` or `0` does not show the pulse-pulse carrier frequency measurement result.

`ON` or `1` shows the pulse-pulse carrier frequency measurement result in the pulse table.

Returns

See Arguments.

Examples

`DISPLAY:PULSE:RESULT:MEASVIEW1:PPFREQUENCY ON` shows the pulse-pulse carrier frequency measurement result in the pulse table.

DISPlay:PULSe:RESult[:MEASview<y>]:PPOWer

Determines whether or not to show the peak power measurement result in the pulse table.

Conditions

Measurement views: Pulse table

Group

Display commands

Syntax

```
DISPlay:PULSe:RESult[:MEASview<y>]:PPOWer { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult[:MEASview<y>]:PPOWer?
```

Arguments

`OFF` or `0` does not show the peak power measurement result.

`ON` or `1` shows the peak power measurement result in the pulse table.

Returns

See Arguments.

Examples

`DISPLAY:PULSE:RESULT:MEASVIEW1:PPOWER ON` shows the peak power measurement result in the pulse table.

DISPlay:PULSe:RESult[:MEASview<y>]:PPPHase

Determines whether or not to show the pulse-pulse carrier phase measurement result in the pulse table.

Conditions

Measurement views: Pulse table

Group

Display commands

Syntax

```
DISPlay:PULSe:RESult:PPPHase { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult[:MEASview<y>]:PPPHase?
```

Arguments

`OFF` or `0` does not show the pulse-pulse carrier phase measurement result.

`ON` or `1` shows the pulse-pulse carrier phase measurement result in the pulse table.

Returns

See Arguments.

Examples

`DISPLAY:PULSE:RESULT:MEASview1:PPPHASE ON` shows the pulse-pulse carrier phase measurement result in the pulse table.

DISPlay:PULSe:RESult[:MEASview<y>]:RINTerval

Determines whether or not to show the repetition interval measurement result in the pulse table.

Conditions

Measurement views: Pulse table

Group

Display commands

Syntax

```
DISPlay:PULSe:RESult[:MEASview<y>]:RINTerval { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult[:MEASview<y>]:RINTerval?
```

Arguments

`OFF` or `0` does not show the repetition interval measurement result.

`ON` or `1` shows the repetition interval measurement result in the results table.

Returns

See Arguments.

Examples

DISPLAY:PULSE:RESULT:MEASVIEW1:RINTERVAL ON shows the repetition interval measurement result in the pulse table.

DISPlay:PULSe:RESult[:MEASview<y>]:RIPDb

Sets or queries showing the ripple measurement result in dB in the pulse table.

Conditions

Measurement views: Pulse table

Group

Display commands

Syntax

```
DISPlay:PULSe:RESult[:MEASview<y>]:RIPDb { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult[:MEASview<y>]:RIPDb?
```

Arguments

OFF or 0 does not show the ripple measurement result in dB.

ON or 1 shows the ripple measurement result in dB in the pulse table.

Returns

See Arguments.

Examples

DISPLAY:PULSE:RESULT:MEASVIEW1:RIPDB ON shows the ripple measurement result in dB in the pulse table.

DISPlay:PULSe:RESult:MEASview<y>:RIPPle

Sets or queries showing the ripple measurement result in the pulse table.

Conditions

Measurement views: Pulse table

Group

Display commands

Syntax

```
DISPlay:PULSe:RESult:MEASview<y>:RIPPle { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult:MEASview<y>:RIPPle?
```

Arguments

OFF or 0 does not show the ripple measurement result.

ON or 1 shows the ripple measurement result in the pulse table.

Returns

See Arguments.

Examples

DISPLAY:PULSE:RESULT:MEASVIEW1:RIPPLE ON shows the ripple measurement result in the pulse table.

DISPlay:PULSe:RESult[:MEASview<y>]:RISE

Determines whether or not to show the rise time measurement result in the pulse table.

Conditions

Measurement views: Pulse table

Group

Display commands

Syntax

```
DISPlay:PULSe:RESult[:MEASview<y>]:RISE { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult[:MEASview<y>]:RISE?
```

Arguments

`OFF` or `0` does not show the rise time measurement result.

`ON` or `1` shows the rise time measurement result in the pulse table.

Returns

See Arguments.

Examples

`DISPLAY:PULSE:RESULT:MEASVIEW1:RISE ON` shows the rise time measurement result in the pulse table.

DISPlay:PULSe:RESult[:MEASview<y>]:RMSFreqerror

Determines whether or not to show the RMS frequency error measurement result in the pulse table.

Conditions

Measurement views: Pulse table

Group

Display commands

Syntax

```
DISPlay:PULSe:RESult[:MEASview<y>]:RMSFreqerror { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult[:MEASview<y>]:RMSFreqerror?
```

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

`OFF` or `0` does not show the RMS frequency error measurement result.

`ON` or `1` shows the RMS frequency error measurement result in the pulse table.

Returns

See Arguments.

Examples

`DISPLAY:PULSE:RESULT:MEASVIEW1:RMSFREQERROR ON` shows the RMS frequency error measurement result in the pulse table.

DISPlay:PULSe:RESult[:MEASview<y>]:RMSPherror

Determines whether or not to show the RMS phase error measurement result in the pulse table.

Conditions

Measurement views: Pulse table

Group

Display commands

Syntax

```
DISPlay:PULSe:RESult[:MEASview<y>]:RMSPherror { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult[:MEASview<y>]:RMSPherror?
```

Arguments

`OFF` or `0` does not show the RMS phase error measurement result.

`ON` or `1` shows the RMS phase error measurement result in the pulse table.

Returns

See Arguments.

Examples

`DISPLAY:PULSE:RESULT:MEASVIEW1:RMSPHERROR ON` shows the RMS phase error measurement result in the pulse table.

DISPlay:PULSe:RESult[:MEASview<y>]:RRATe

Determines whether or not to show the repetition rate measurement result in the pulse table.

Conditions

Measurement views: Pulse table

Group

Display commands

Syntax

```
DISPlay:PULSe:RESult[:MEASview<y>]:RRATe { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult[:MEASview<y>]:RRATe?
```

Arguments

`OFF` or `0` does not show the repetition rate measurement result.

`ON` or `1` shows the repetition rate measurement result in the pulse table.

Returns

See Arguments.

Examples

`DISPLAY:PULSE:RESULT:MEASVIEW1:RRATE ON` shows the repetition rate measurement result in the pulse table.

DISPlay:PULSe:RESult[:MEASview<y>]:TIME

Determines whether or not to show the time measurement result in the pulse table.

Conditions

Measurement views: Pulse table

Group

Display commands

Syntax

```
DISPlay:PULSe:RESult[:MEASview<y>]:TIME { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult[:MEASview<y>]:TIME?
```

Arguments

`OFF` or `0` does not show the time measurement result.

`ON` or `1` shows the time measurement result in the pulse table.

Returns

See Arguments.

Examples

`DISPLAY:PULSE:RESULT:MEASVIEW1:TIME ON` shows the time measurement result in the pulse table.

DISPlay:PULSe:RESult[:MEASview<y>]:WIDTh

Determines whether or not to show the pulse width measurement result in the pulse table.

Conditions

Measurement views: Pulse table

Group

Display commands

Syntax

```
DISPlay:PULSe:RESult[:MEASview<y>]:WIDTh { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult[:MEASview<y>]:WIDTh?
```

Arguments

`OFF` or `0` does not show the pulse width measurement result.

`ON` or `1` shows the pulse width measurement result in the pulse table.

Returns

See Arguments.

Examples

`DISPLAY:PULSE:RESULT:MEASVIEW1:WIDTH ON` shows the pulse width measurement result in the pulse table.

DISPlay:PULSe[:MEASview<y>]:SELect:NUMBer

Sets or queries a pulse to measure. For the selected pulse, the statistics view indicates the measurement result while the table view highlights it, and the trace view displays the waveform.

Conditions

Measurement views: : Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Display commands

Syntax

```
DISPlay:PULSe[:MEASview<y>]:SELect:NUMBer <number>
DISPlay:PULSe[:MEASview<y>]:SELect:NUMBer?
```

Arguments

`<number>::<NR1>` specifies the number of pulse to measure.Range: -(the number of acquired pulses before the time reference) to +(the number of acquired pulses after the time reference).

Zero (0) represents the pulse at the analysis time reference specified using the `[SENSe]:ANALysis:REFerence` command. The number of acquired pulses depends on the analysis range.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<NR1>`

Examples

`DISPlay:PULSe:MEASview2:SELect:NUMBer -28` measures the pulse #-28.

DISPlay:PULSe[:MEASview<y>]:SELect:RESult

Sets or queries which result is shown in the pulse trace and statistics views.

Conditions

Measurement views: Pulse statistics, Pulse trace

Group

Display commands

Syntax

```
DISPlay:PULSe[:MEASview<y>]:SELect:RESult { AVERage | PPOWer | ATX |WIDTh| RISE | FALL | RINTerval
    | RRATe | DUTPct | DUTRatio |RIPPle| RIPDb | DROop | DRODb | OVERshoot | OVEDb | PPPHase|
    PPFRequency | RMSFreqerror | MFReqerror| RMSPherror |MPHerror | FRDeviation | FDELta |
    PHDeviation | IRAMplitude| IRTime }
DISPlay:PULSe[:MEASview<y>]:SELect:RESult?
```

Arguments

The following table lists the arguments.

Pulse results

Argument	Result
AVERage	Average on power
PPOWer	Peak power
ATX	Average transmitted power
WIDTh	Pulse width
RISE	Rise time
FALL	Fall time
RINTerval	Repetition interval
RRATe	Repetition rate
DUTPct	Duty factor (%)
DUTRatio	Duty factor (ratio)
RIPPle	Ripple
RIPDb	Ripple (dB)
DROop	Droop
DRODb	Droop (dB)
OVERshoot	Overshoot
OVEDb	Overshoot (dB)
PPPHase	Pulse-pulse carrier phase
PPFRequency	Pulse-pulse carrier frequency
RMSFreqerror	RMS frequency error
MFReqerror	Maximum frequency error
RMSPherror	RMS phase error
MPHerror	Maximum phase error
FRDeviation	Frequency deviation
PHDeviation	Phase deviation
FDELta	Delta frequency

Argument	Result
IRAMplitude	Impulse Response Amplitude
IRTime	Impulse Response Sidelobe Offset Time from Mainlobe

Returns

See Arguments.

Examples

`DISPlay:PULSe:MEASview1:SELect:RESult AVERage` shows the average on power result in the pulse trace and statistics views.

DISPlay:PULSe[:MEASview<y>]:SELect:RESult

Sets or queries the result shown in the pulse trace and statistics views.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse trace with specified Measurement view number

Group

Display commands

Syntax

DISPlay:PULSe[:MEASview<y>]:SELect:RESult <arg>

DISPlay:PULSe[:MEASview<y>]:SELect:RESult?

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

AVERage is the average on power

PPOWer is the peak power

ATX is the average transmitted power

WIDTh is the Pulse width

RISE is the rise time

FALL is the fall time

RINTerval is the repetition interval

RRATe is the repetition rate

DUTPct is the duty factor (%)

DUTRatio is the duty factor (ratio)

RIPPle is the ripple

RIPDb is the ripple (dB)

DROop is the droop

DRODb is the droop (dB)

OVERshoot is the overshoot

OVEDb is the overshoot (dB)

PPPHase is the pulse-pulse carrier phase

PPFRequency is the pulse-pulse carrier frequency

RMSFreqerror is the RMS frequency error

MFReqerror is the maximum frequency error

RMSPherror is the RMS phase error

MPHerror is the maximum phase error

FRDeviation is the frequency deviation

FDELta is the delta frequency

PHDeviation is the phase deviation

IRAMplitude is the impulse response amplitude

TIME is the time in seconds

IRTime is the impulse response sidelobe offset time from the mainlobe

PPPD is the pulse to pulse phase difference

PPFD is the pulse to pulse frequency difference

FABS is the absolute frequency

Returns

<NRf>

Examples

DISPlay:PULSe:MEASview1:SELect:RESult AVERAGE shows the average on power result in the pulse trace and statistics views.

DISPlay:PULSe:STATistics[:MEASview<y>]:MARKer<x>:SHOW:STATe

Determines whether to show or hide the marker readout in the statistics graph. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to FFT.

Conditions

Measurement views: Pulse statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:STATistics[:MEASview<y>]:MARKer<x>:SHOW:STATe { OFF | ON | 0 | 1 }
DISPlay:PULSe:STATistics[:MEASview<y>]:MARKer<x>:SHOW:STATe?
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<x> = 0 to 4.

`OFF` or 0 hides the marker readout.

ON or 1 shows the marker readout.

Returns

See Arguments.

Examples

DISPlay:PULSe:STATistics:MEASview1:MARKer:SHOW:STATe ON shows the marker readout in the statistics graph.

DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT

Sets or queries how to show the statistics graph.

Conditions

Measurement views: Pulse statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT { TRENd | FFT | TTRend | HISTogram }
DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT?
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<x> = 0 to 4.

TRENd shows the statistics result along with the pulse number.

FFT shows the statistics result transformed into the frequency domain by FFT.

TTRend shows the statistics result along with time.

HISTogram shows the histogram statistics results.

Returns

See Arguments.

Examples

DISPLAY:PULSE:STATISTICS:MEASview1:PLOT TRENd shows the statistics result along with the pulse number.

DISPlay:PULSe:STATistics[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid in the statistics view.

Conditions

Measurement views: Pulse statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:STATistics[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:PULSe:STATistics[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or `0` hides the graticule grid.

`ON` or `1` shows the graticule grid.

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Examples

`DISPLAY:PULSE:STATISTICS:MEASVIEW2:WINDOW:TRACE:GRATICULE:GRID:STATE ON` shows the graticule grid on the Pulse Statistics.

DISPlay:PULSe:STATistics[:MEASview<y>]:X:RSCale (No Query Form)

Rescales the horizontal axis to fit the waveform to the screen in the statistics graph.

Conditions

Measurement views: Pulse statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:STATistics[:MEASview<y>]:X:RSCale
```

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Examples

`DISPlay:PULSe:STATistics:MEASview2:X:RSCale` rescales the horizontal axis of the statistics graph.

DISPlay:PULSe:STATistics[:MEASview<y>]:X[:SCALe]:NUMBer

Sets or queries the horizontal full scale in the statistics graph.

Conditions

Measurement views: Pulse statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:STATistics[:MEASview<y>]:X[:SCALe]:NUMBer <value>
DISPlay:PULSe:STATistics[:MEASview<y>]:X[:SCALe]:NUMBer?
```

Related Commands

DISPlay:PULSe:STATistics[:MEASview<y>]:X[:SCALe]:OFFSet (on page 942)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

<value>::<NRf> specifies the horizontal full scale. The setting range depends on the DISPlay:PULSe:STATistics:PLOT command parameters as shown in the table below.

DISPlay:PULSe:STATistics:PLOT	Setting range
TRENd	1 to 1000
FFT	1 Hz to 120 MHz

Returns

<NRf>

Examples

DISPlay:PULSe:STATistics:MEASview2:X:SCALe:NUMBer 50 sets the horizontal full scale to 50 pulses when the plot is trend.

DISPlay:PULSe:STATistics[:MEASview<y>]:X:SCALe:OFFSet

Sets or queries the minimum horizontal value (the first pulse to show) in the statistics graph.

Conditions

Measurement views: Pulse statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:STATistics[:MEASview<y>]:X:SCALe:OFFSet <value>
DISPlay:PULSe:STATistics[:MEASview<y>]:X:SCALe:OFFSet?
```

Arguments

<value>::<NRf> specifies the number of the first pulse.

Range: -(X - X/10) to +(X - X/10) where X is the horizontal scale set by the DISPlay:PULSe:STATistics:X[:SCALe]:NUMBer command.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf>

Examples

DISPLAY:PULSE:STATISTICS:MEASview2:X:SCALE:OFFSET 120 sets the first pulse number to #120.

DISPlay:PULSe:STATistics[:MEASview<y>]:Y:RSCale (No Query Form)

Rescales the vertical axis to fit the waveform to the screen in the statistics graph.

Conditions

Measurement views: Pulse statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:STATistics[:MEASview<y>]:Y:RSCale
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Examples

`DISPlay:PULSe:STATistics:MEASview2:Y:RSCale` rescales the vertical axis of the statistics graph.

DISPlay:PULSe:STATistics[:MEASview<y>]:Y[:SCALe]:FULL

Sets or queries the vertical full scale in the statistics graph.

Conditions

Measurement views: Pulse statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:STATistics[:MEASview<y>]:Y[:SCALe]:FULL <value>
DISPlay:PULSe:STATistics[:MEASview<y>]:Y[:SCALe]:FULL
```

Related Commands

DISPlay:PULSe:STATistics[:MEASview<y>]:X[:SCALe]:OFFSET (on page 944)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

<value>::<NRf> specifies the vertical full scale. The setting range depends on the `DISPlay:PULSe:STATistics:PLOT` command parameters and the measurement items as shown in the following table.

DISPlay:PULSe:STATistics:PLOT	Measurement item	Setting range
TRENd	Average on power, Peak power, Average transmitted power	0.1 to 200 dB
	Pulse width, Rise time, Fall time, Repetition interval	1 n to 5 Ms
	Repetition rate	100 m to 100 MHz
	Duty factor, Ripple	1 to 100%
	Droop	1 to 200%
	Pulse-pulse carrier phase	1 to 360°
FFT	All	10 to 200 dB

Returns

<NRf>

Examples

DISPlay:PULSe:STATistics:MEASview2:Y:SCALe:FULL 100 sets the vertical full scale to 100 dB.

DISPlay:PULSe:STATistics[:MEASview<y>]:Y[:SCALe]:OFFSet

Sets or queries the vertical offset in the statistics graph.

Conditions

Measurement views: Pulse statistics

Group

Display commands

Syntax

```
DISPlay:PULSe:STATistics[:MEASview<y>]:Y[:SCALe]:OFFSet <value>
DISPlay:PULSe:STATistics[:MEASview<y>]:Y[:SCALe]:OFFSet?
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

<value>::<NRf> specifies the vertical offset. The vertical offset is the value at the top or the bottom edge of the graph depending on measurement items. The setting range depends on the DISPlay:PULSe:STATistics:PLOT command parameters and the measurement items as shown in the table below.

DISPlay:PULSe:STATistics:PLOT	Measurement item	Setting range	Offset position [1]
TRENd	Average on power, Peak power, Average transmitted power	-170 to +50 dBm	Top (Bottom for the unit of Volts or Watts)
	Pulse width, Rise time, Fall time, Repetition interval	0 to 5 Ms	Bottom
	Repetition rate	0 to 100 MHz	Bottom
	Duty factor, Ripple	0 to 100%	Bottom
	Droop	0 to +100%	Bottom
	Pulse-pulse carrier phase	-180 to +180°	Bottom
FFT	All	-400 to +100 dB	Top

Returns

<NRf>

Examples

DISPLAY:PULSE:STATISTICS:MEASview2:Y:SCALE:OFFSET 24.8 sets the maximum vertical value to 24.8 dBm in the statistics graph.

DISPlay:PULSe:STATistics[:MEASview<y>]:Y[:SCALe]:STOP? (Query Only)

Queries the minimum vertical value (bottom edge) in the statistics graph.

Conditions

Measurement views: Pulse statistics

Group

Display commands

Syntax

DISPlay:PULSe:STATistics[:MEASview<y>]:Y[:SCALe]:STOP?

Related Commands

DISPlay:PULSe:STATistics[:MEASview<y>]:Y[:SCALe]:OFFSET (on page 944)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<y_stop>::<NRf> is the minimum vertical value (bottom edge).

Examples

DISPlay:PULSe:STATistics:MEASview2:Y:SCALe:STOP? might return –150.0, indicating that the minimum vertical value is -150 dBm in the pulse statistics graph.

DISPlay:PULSe:TRACe[:MEASview<y>]:MARKer:SHOW:STATe

Determines whether to show or hide the marker readout in the pulse trace view.

Conditions

Measurement views: Pulse trace with specified Measurement view number

Group

Display commands

Syntax

```
DISPlay:PULSe:TRACe[:MEASview<y>]:MARKer:SHOW:STATe { OFF | ON | 0 | 1 }
DISPlay:PULSe:TRACe[:MEASview<y>]:MARKer:SHOW:STATe?
```

Arguments

OFF or 0 hides the marker readout.

ON or 1 shows the marker readout.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Examples

DISPLAY:PULSE:TRACE:MEASVIEW1:MARKER:SHOW:STATE ON shows the marker readout on the pulse trace measurement view 1.

DISPlay:PULSe:TRACe[:MEASview<y>]:POINt:SHOW

Determines whether to show or hide the measurement points and lines in the pulse trace view.

Conditions

Measurement views: Pulse trace with specified Measurement view number

Group

Display commands

Syntax

```
DISPlay:PULSe:TRACe[:MEASview<y>]:POINt:SHOW { OFF | ON | 0 | 1 }
DISPlay:PULSe:TRACe[:MEASview<y>]:POINt:SHOW?
```

Arguments

OFF or 0 hides the measurement points and lines.

ON or 1 shows the measurement points and lines.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Examples

DISPLAY:PULSE:TRACE:MEASVIEW1:POINT:SHOW ON shows the measurement points and lines in the pulse trace measurement view 1.

DISPlay:PULSe:TRACe[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid in the pulse trace view.

Conditions

Measurement views: Pulse trace with specified Measurement view number

Group

Display commands

Syntax

```
DISPlay:PULSe:TRACe[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:PULSe:TRACe[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

OFF or 0 hides the graticule grid.

ON or 1 shows the graticule grid.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Examples

DISPLAY:PULSE:TRACE:MEASVIEW1:WINDOW:TRACE:GRATICULE:GRID:STATE ON shows the graticule grid on the pulse trace measurement view 1.

DISPlay:PULSe:TRACe[:MEASview<y>]:X:RSCale

Rescales the horizontal axis to fit the waveform to the screen in the pulse trace view.

Conditions

Measurement views: Pulse trace with specified Measurement view number

Group

Display commands

Syntax

```
DISPlay:PULSe:TRACe[:MEASview<y>]:X:RSCale
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Examples

DISPLAY:PULSE:TRACE:MEASVIEW1:X:RSCALE rescales the horizontal axis in the pulse trace Measurement View 1.

DISPlay:PULSe:TRACe[:MEASview<y>]:X[:SCALe]

Sets or queries the horizontal full scale in the pulse trace view.

Conditions

Measurement views: Pulse trace with specified Measurement view number

Group

Display commands

Syntax

```
DISPlay:PULSe:TRACe[:MEASview<y>]:X[:SCALe] <value>
DISPlay:PULSe:TRACe[:MEASview<y>]:X[:SCALe]?
```

Arguments

<value>::<NRf> specifies the horizontal full scale.

Range: 10 ns to acquisition memory capacity.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf>

Examples

DISPLAY:PULSE:TRACE:MEASVIEW1:X:SCALE 5.5E-6 sets the horizontal scale to 5.5 µs in the pulse trace measurement view 1.

DISPlay:PULSe:TRACe[:MEASview<y>]:X[:SCALe]:FULL

Sets or queries the full-scale reference for the horizontal rescale.

Conditions

Measurement views: Pulse trace with specified Measurement view number

Group

Display commands

Syntax

```
DISPlay:PULSe:TRACe[:MEASview<y>]:X[:SCALe]:FULL { SELected | MAXimum }
DISPlay:PULSe:TRACe[:MEASview<y>]:X[:SCALe]:FULL?
```

Arguments

SELected uses the selected pulse for the full-scale reference.

MAXimum uses the maximum pulse for the full-scale reference.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf>

Examples

DISPLAY:PULSE:TRACE:MEASVIEW1:X:SCALE:FULL SELected uses the selected pulse for the full-scale reference in the pulse trace measurement view 1.

DISPlay:PULSe:TRACe[:MEASview<y>]:X[:SCALe]:OFFSet

Sets or queries the minimum horizontal value (left edge) in the pulse trace view.

Conditions

Measurement views: Pulse trace with specified Measurement view number

Group

Display commands

Syntax

```
DISPlay:PULSe:TRACe[:MEASview<y>]:X[:SCALe]:OFFSet <value>
DISPlay:PULSe:TRACe[:MEASview<y>]:X[:SCALe]:OFFSet?
```

Related Commands

DISPlay:PULSe:TRACe[:MEASview<y>]:X[:SCALe]:PDIVision (on page 949)

Arguments

<value>::<NRf> specifies the minimum horizontal value.

Range: [(analysis offset) - (X scale) × 0.9] to [(analysis offset) + (analysis length) - (X scale) × 0.1]

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf>

Examples

DISPLAY:PULSE:TRACE:MEASVIEW1:X:SCALE:OFFSET 937.5E-9 sets the minimum horizontal value to 937.5 ns in the pulse trace measurement view 1.

DISPlay:PULSe:TRACe[:MEASview<y>]:X[:SCALe]:PDIVision

Sets or queries the horizontal full scale in the pulse trace view.

Conditions

Measurement views: Pulse trace with specified Measurement view number

Group

Display commands

Syntax

```
DISPlay:PULSe:TRACe[:MEASview<y>]:X[:SCALe]:PDIVision <value>
DISPlay:PULSe:TRACe[:MEASview<y>]:X[:SCALe]:PDIVision?
```

Arguments

<value>::<NRf> specifies the horizontal full scale.

Range: 10 ns to acquisition memory capacity.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf>

Examples

DISPLAY:PULSE:TRACE:MEASVIEW1:X:SCALE:PDIVISION 5.5E-6 sets the horizontal scale to 5.5 µs in the pulse trace measurement view 1.

DISPlay:PULSe:TRACe[:MEASview<y>]:Y:RSCale

Rescales the vertical axis to fit the waveform to the screen in the pulse trace view.

Conditions

Measurement views: Pulse trace with specified Measurement view number

Group

Display commands

Syntax

DISPlay:PULSe[:MEASview<y>]:TRACe:Y:RSCale

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Examples

DISPLAY:PULSE:TRACE:MEASVIEW1:Y:RSCALE rescales the vertical axis in the pulse trace measurement view 1.

DISPlay:PULSe:TRACe[:MEASview<y>]:Y[:SCALe]:FULL

Sets or queries the vertical full scale in the pulse trace view.

Conditions

Measurement views: Pulse trace with specified Measurement view number

Group

Display commands

Syntax

DISPlay:PULSe:TRACe[:MEASview<y>]:Y[:SCALe]:FULL <value>
DISPlay:PULSe:TRACe[:MEASview<y>]:Y[:SCALe]:FULL?

Related Commands

DISPlay:PULSe:TRACe[:MEASview<y>]:Y[:SCALe]:OFFSet (on page 951)

Arguments

<value>::<NRf> specifies the vertical full scale.

Range: 0.1 to 200 dB.

The unit can be changed by the [SENSe]:POWer:UNITs command.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf>

Examples

DISPLAY:PULSE:TRACE:MEASVIEW1:Y:SCALE:FULL 100 sets the vertical full scale to 100 dB on the pulse trace measurement view 1.

DISPlay:PULSe:TRACe[:MEASview<y>]:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (the value at the top edge of the vertical axis) in the pulse trace view.

Conditions

Measurement views: Pulse trace with specified Measurement view number

Group

Display commands

Syntax

```
DISPlay:PULSe:TRACe[:MEASview<y>]:Y[:SCALe]:OFFSet <value>
DISPlay:PULSe:TRACe[:MEASview<y>]:Y[:SCALe]:OFFSet?
```

Related Commands

DISPlay:PULSe:TRACe[:MEASview<y>]:Y[:SCALe]:STOP? (on page 951)

Arguments

<value>::<NRf> specifies the vertical offset. Range: -170 to +50 dBm.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf>

Examples

DISPLAY:PULSE:TRACE:MEASVIEW1:Y:SCALE:OFFSET 23.5 sets the vertical offset to 23.5 dBm on the pulse trace measurement view 1.

DISPlay:PULSe:TRACe[:MEASview<y>]:Y[:SCALe]:STOP

Queries the minimum vertical value (bottom edge) in the pulse trace view.

Conditions

Measurement views: Pulse trace with specified Measurement view number

Group

Display commands

Syntax

```
DISPlay:PULSe:TRACe:MEASview<y>:Y[:SCALe]:STOP?
```

Related Commands

DISPlay:PULSe:TRACe[:MEASview<y>]:Y[:SCALe]:OFFSet (on page 951)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<y_stop>::<NRf> is the minimum vertical value (bottom edge).

Examples

DISPLAY:PULSE:TRACE:MEASVIEW1:Y:SCALE:STOP? might return -150.0, indicating that the minimum vertical value is -150 dBm in the pulse trace measurement view 1.

DISPlay:SEM:MARKer:SHOW:STATe

Determines whether to show or hide the readout for the selected marker in the Spectral Emissions Mask view.

Conditions

Measurement views: Spectral Emissions Mask

Group

Display commands

Syntax

```
DISPlay:SEM:MARKer:SHOW:STATe { OFF | ON | 1| 0 }
DISPlay:SEM:MARKer:SHOW:STATe?
```

Arguments

OFF or 0 hides the readout for the selected marker in the graph.

ON or 1 shows the readout for the selected marker in the graph.

Examples

DISPLAY:SEM:MARKER:SHOW:STATE ON shows the readout for the selected marker in the graph.

DISPlay:SEM:RESet:SCALe (No Query Form)

Resets the scale in the Spectral Emissions Mask view.

Conditions

Measurement views: Spectral Emissions Mask

Group

Display commands

Syntax

```
DISPlay:SEM:RESet:SCALe
```

Arguments

None

Examples

`DISPLAY:SEM:RESET:SCALE` resets the scale in the view.

DISPlay:SEM:SHOW:LIMit

Sets or queries the appearance of the limits in the Spectral Emissions Mask view

Conditions

Measurement views: Spectral Emissions Mask

Group

Display commands

Syntax

```
DISPlay:SEM:SHOW:LIMit { SHADed | LINE | OFF }
DISPlay:SEM:SHOW:LIMit?
```

Arguments

`SHADed` uses shading for the limit area in the display.

`LINE` uses a line for the limit area.

`OFF` turns off the limits in the display.

Examples

`DISPLAY:SEM:SHOW:LIMIT SHADed` uses shading to identify the limits in the display.

DISPlay:SEM:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule in the Spectral Emissions Mask view.

Conditions

Measurement views: Spectral Emissions Mask

Group

Display commands

Syntax

```
DISPlay:SEM:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 1| 0}
DISPlay:SEM:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or `0` hides the graticule in the graph.

`ON` or `1` shows the graticule in the graph.

Examples

`DISPLAY:SEM:WINDOW:TRACE:GRATICULE:GRID:STATE OFF` turns off the graticule in the graph.

DISPlay:SEM:X[:SCALe]:AUTO (No Query Form)

Rescales the horizontal scale automatically to fit the waveform in the Spectral Emissions Mask view.

Conditions

Measurement views: Spectral Emissions Mask

Group

Display commands

Syntax

```
DISPlay:SEM:X[:SCALe]:AUTO
```

Arguments

None

Examples

`DISPLAY:SEM:X[:SCALE]:AUTO` rescales the horizontal scale to fit the waveform on the screen.

DISPlay:SEM:X[:SCALe]:START

Sets or queries the start frequency (left edge) of the Spectral Emissions Mask graph.

Conditions

Measurement views: Spectral Emissions Mask

Group

Display commands

Syntax

```
DISPlay:SEM:X[:SCALe]:START <value>
DISPlay:SEM:X[:SCALe]:START?
```

Arguments

`<value>::<NRf>` specifies the start frequency.

Examples

`DISPLAY:SEM:X[:SCALE]:START 10Hz` sets the start frequency to 10 Hz in the graph.

DISPlay:SEM:X[:SCALe]:STOP

Sets or queries the stop frequency (right edge) of the Spectral Emissions Mask graph.

Conditions

Measurement views: Spectral Emissions Mask

Group

Display commands

Syntax

```
DISPlay:SEM:X[:SCALe]:STOP <value>
DISPlay:SEM:X[:SCALe]:STOP?
```

Arguments

`<value>::<NRf>` specifies the start frequency.

Examples

`DISPLAY:SEM:X[:SCALE]:STOP 2GHz` sets the stop frequency to 2 GHz in the graph.

DISPlay:SEM:Y[:SCALe]

Sets or queries the vertical range of the Spectral Emissions Mask view.

Conditions

Measurement views: Spectral Emissions Mask

Group

Display commands

Syntax

```
DISPlay:SEM:Y[:SCALe] <value>
DISPlay:SEM:Y[:SCALe]?
```

Arguments

`<value>::= <NRf>` specifies the vertical scale.

Examples

`DISPLAY:SEM:Y[:SCALE] 200` sets the vertical scale to 200 MHz.

DISPlay:SEM:Y[:SCALe]:AUTO (No Query Form)

Rescales the vertical scale automatically to fit the waveform in the Spectral Emissions Mask view.

Conditions

Measurement views: Spectral Emissions Mask

Group

Display commands

Syntax

```
DISPlay:SEM:Y[:SCALe]:AUTO
```

Arguments

None

Examples

`DISPLAY:SEM:Y[:SCALE]:AUTO` rescales the vertical scale to fit the waveform on the screen.

DISPlay:SEM:Y[:SCALe]:OFFSet

Sets or queries the vertical offset in the Spectral Emissions Mask view.

Conditions

Measurement views: Spectral Emissions Mask

Group

Display commands

Syntax

```
DISPlay:SEM:Y[:SCALe]:OFFSet <value>
DISPlay:SEM:Y[:SCALe]:OFFSet?
```

Arguments

`<value>::= <NRf>` specifies the horizontal range.

Examples

`DISPLAY:SEM:Y[:SCALE]:OFFSET 40` sets the vertical position to 40 dBm.

DISPlay:SGRam[:MEASview<y>]:FREQuency:AUTO (No Query Form)

Rescales the horizontal axis automatically to fit the waveform to the screen in the spectrogram view.

Conditions

Measurement views: Spectrogram

Group

Display commands

Syntax

```
DISPlay:SGRam[:MEASview<y>]:FREQuency:AUTO
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Examples

`DISPlay:SGRam:MEASview1:FREQuency:AUTO` rescales the horizontal scale automatically to fit the waveform to the screen in the spectrogram measurement view 1.

DISPlay:SGRam[:MEASview<y>]:FREQuency:OFFSet

Sets or queries the frequency offset (the value at the center of the horizontal axis) in the spectrogram.

Conditions

Measurement views: Spectrogram

Group

Display commands

Syntax

```
DISPlay:SGRam[:MEASview<y>]:FREQuency:OFFSet <value>
DISPlay:SGRam[:MEASview<y>]:FREQuency:OFFSet?
```

Related Commands

DISPlay:SGRam[:MEASview<y>]:FREQuency[:SCAle] (on page 957)

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

`<value>::<NRf>` specifies the frequency offset.

Range: [(center frequency) - (X scale) × 0.9] to [(center frequency) + (X scale) × 0.9]

Examples

`DISPlay:SGRam:MEASview1:FREQuency:OFFSet 1.45GHz` sets the frequency offset to 1.45 GHz in the spectrogram measurement view 1.

DISPlay:SGRam[:MEASview<y>]:FREQuency[:SCAle]

Sets or queries the horizontal range of the spectrogram.

Conditions

Measurement views: Spectrogram

Group

Display commands

Syntax

```
DISPlay:SGRam[:MEASview<y>]:FREQuency[:SCAle] <value>
DISPlay:SGRam[:MEASview<y>]:FREQuency[:SCAle]?
```

Related Commands

DISPlay:SGRam[:MEASview<y>]:FREQuency:OFFSet (on page 956)

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

`<value>::<NRf>` specifies the horizontal range.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Returns

<NRf> Returns the horizontal range

Examples

`DISPlay:SGRam:MEASview1:FREQuency:SCAle 10MHz` sets the horizontal range to 10 MHz on the trace of Spectrogram measurement view 1.

DISPlay:SGRam[:MEASview<y>]:MARKer:SHOW:STATe

Sets or queries the display of the Marker legend in the Spectrogram graph.

Conditions

Measurement views: Spectrogram

Group

Display commands

Syntax

```
DISPlay:SGRam[:MEASview<y>]:MARKer:SHOW:STATe{ OFF | ON | 0 | 1 }
DISPlay:SGRam[:MEASview<y>]:MARKer:SHOW:STATe?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

`OFF` or 0 removes or turns off the Marker legend.

`ON` or 1 displays or turns on the Marker legend.

Returns

{ OFF | ON | 0 | 1 }

Examples

`DISPlay:SGRam:MEASview1:MARKer:SHOW:STATe ON` turns on the display of the Marker legend in the spectrogram graph of view 1.

DISPlay:SGRam[:MEASview<y>]:SELected:TIMestamp

Sets or queries the display of the time stamp in the Spectrogram graph.

Conditions

Measurement views: Spectrogram

Group

Display commands

Syntax

```
DISPlay:SGRam[:MEASview<y>]:SELected:TIMestamp {OFF | ON | 0 | 1}
DISPlay:SGRam[:MEASview<y>]:SELected:TIMestamp?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

`OFF` or `0` removes or turns off the time stamp.

`ON` or `1` displays or turns on the time stamp.

Returns

{OFF | ON | 0 | 1}

Examples

`DISPlay:SGRam:MEASview1:SELected:TIMestamp?` queries the display status of the time stamp in the spectrogram graph view 1.

DISPlay:SGRam[:MEASview<y>]:TIME:AUTO (No Query Form)

Rescales the vertical axis automatically to fit the graph to the screen in the spectrogram view.

Conditions

Measurement views: Spectrogram

Group

Display commands

Syntax

`DISPlay:SGRam[:MEASview<y>]:TIME:AUTO`

Arguments

`<y>` = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Examples

`DISPlay:SGRam:MEASview1:TIME:AUTO` rescales the vertical scale automatically to fit the waveform to the screen in the spectrogram measurement view 1.

DISPlay:SGRam[:MEASview<y>]:TIME:OFFSet

Sets or queries the vertical axis (time) offset (bottom line number) in the spectrogram.

Conditions

Measurement views: Spectrogram

Group

Display commands

Syntax

```
DISPlay:SGRam[:MEASview<y>]:TIME:OFFSet <value>
DISPlay:SGRam[:MEASview<y>]:TIME:OFFSet?
```

Related Commands

DISPlay:SGRam[:MEASview<y>]:TIME[:SCAle] (on page 961)

Arguments

`<y>` = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

`<value>::<NRf>` specifies the time offset.

Range: Line #0 to 125000. Zero (0) represents the latest line.

Returns

<NRf> specifies the time offset

Examples

`DISPlay:SGRam:MEASview1:TIME:OFFSet 15` sets the time offset to Line #15 in the spectrogram view 1.

DISPlay:SGRam[:MEASview<y>]:TIME:OFFSet:DIVisions

Sets or queries the Vertical offset or Position in divisions.

Conditions

Measurement views: Spectrogram

Group

Display commands

Syntax

```
DISPlay:SGRam[:MEASview<y>]:TIME:OFFSet:DIVisions <value>
DISPlay:SGRam[:MEASview<y>]:TIME:OFFSet:DIVisions?
```

Arguments

`<y>` = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

`<value>::<NRf>` sets the vertical offset value in divisions. Range is -130,000 to +130,000.

Returns

<NRf> specifies the vertical offset

Examples

`DISPlay:SGRam:MEASview1:TIME:OFFSet:DIVisions 10` sets the vertical offset to 10 divisions in the spectrogram view 1.

DISPlay:SGRam[:MEASview<y>]:TIME:OVERlap

Sets or queries if overlap is enabled or not.

Conditions

Measurement view: Spectrogram

Group

Display commands

Syntax

```
DISPlay:SGRam[:MEASview<y>]:TIME:OVERlap { OFF | ON | 0 | 1 }
DISPlay:SGRam[:MEASview<y>]:TIME:OVERlap?
```

Arguments

`<y>` = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

`OFF` or `0` hides the overlap in the spectrogram view.

ON or 1 shows the overlap in the spectrogram view.

Returns

{ OFF | ON | 0 | 1 }

Examples

DISPlay:SGRam:MEASview1:TIME:OVERlap ON enables time overlap for spectrogram.

DISPlay:SGRam[:MEASview<y>]:TIME:OVERlap:PERCent

Sets or queries the Overlap in percent. Note that changing this value may change the time per division DISPlay:SGRam:MEASview<y>:TIME:SCAle:PER:DIVision value.

Conditions

Measurement views: Spectrogram

Group

Display commands

Syntax

```
DISPlay:SGRam[:MEASview<y>]:TIME:OVERlap:PERCent <value>
DISPlay:SGRam[:MEASview<y>]:TIME:OVERlap:PERCent?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

<value>::<NRf> specifies the overlap in percent.

Returns

<NRf> specifies the overlap in percent

Examples

DISPlay:SGRam:MEASview1:TIME:OVERlap:PERCent 1.0E1 sets the overlap to 10% in the spectrogram view 1.

DISPlay:SGRam[:MEASview<y>]:TIME[:SCAle]

NOTE: The user interface on the instrument no longer presents this setting. However, the command is still function. Setting this value will change the TIME:PER:DIVision value.

Sets or queries the vertical scale (the amount of time in each line) in the spectrogram. The vertical axis is composed of successive spectral displays. The new spectra can be added at a timed rate specified by this command. For example, if you set the scale to -5, one line is displayed every 5 spectra.

Conditions

Measurement views: Spectrogram

Group

Display commands

Syntax

```
DISPlay:SGRam[:MEASview<y>]:TIME[:SCAle] <value>
DISPlay:SGRam[:MEASview<y>]:TIME[:SCAle]?
```

Related Commands

DISPlay:SGRam[:MEASview<y>]:TIME:OFFSet (on page 959)

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

<value>::<NR1> specifies the vertical scale.

Range: -1023 to 0. Zero (0) displays every spectrum.

Returns

<NRf> specifies the vertical scale

Examples

`DISPlay:SGRam:MEASview1:TIME:SCAle -5` displays one line every 5 spectra in the spectrogram view 1.

DISPlay:SGRam[:MEASview<y>]:TIME[:SCAle]:PER:DIVision

Sets or queries the vertical Time/division value. Note that changing this value may change the `DISPlay:SGRam:MEASview<y>:TIME:OVERlap:PERCent` value.

Conditions

Measurement views: Spectrogram

Group

Display commands

Syntax

```
DISPlay:SGRam[:MEASview<y>]:TIME[:SCAle]:PER:DIVision <value>
DISPlay:SGRam[:MEASview<y>]:TIME[:SCAle]:PER:DIVision?
```

Related Commands

DISPlay:SGRam[:MEASview<y>]:TIME:OVERlap:PERCent (on page 961)

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

<value>::<NRf> specifies the time per division in seconds.

Returns

<NRf> specifies the time per division in seconds

Examples

`DISPlay:SGRam:MEASview1:TIME:SCAle:PER:DIVision?` might return `1.1160000000E-3` in the spectrogram view 1.

DISPlay:SGRam[:MEASview<y>]:TIME:SPECtrums:PERLine (Query Only)

Queries the Spectrums/Line value.

Conditions

Measurement views: Spectrogram

Group

Display commands

Syntax

`DISPlay:SGRam[:MEASview<y>]:TIME:SPECtrums:PERLine?`

Arguments

`<y>` = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Returns

<NRf> specifies the specturm/Line value

Examples

`DISPlay:SGRam:MEASview1:TIME:SPECtrums:PERLine?` might return a `1.0000000000` as a response in the spectrogram view 1.

DISPlay:SPECtrum[:MEASview<y>]:FREQuency:AUTO (No Query Form)

Rescales the horizontal axis automatically to fit the waveform to the screen in the spectrum view.

Conditions

Measurement views: Spectrum

Group

Display commands

Syntax

`DISPlay:SPECtrum[:MEASview<y>]:FREQuency:AUTO`

Arguments

The parameter `<y>`=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`DISPlay:SPECtrum:MEASview2:FREQuency:AUTO` rescales the horizontal scale automatically to fit the waveform to the screen.

DISPlay:SPECtrum[:MEASview<y>]:FREQuency:OFFSet

Sets or queries the frequency offset (the value at the center of the horizontal axis) in the spectrum graph.

Conditions

Measurement views: Spectrum

Group

Display commands

Syntax

```
DISPlay:SPECtrum[:MEASview<y>]:FREQuency:OFFSet <value>
DISPlay:SPECtrum[:MEASview<y>]:FREQuency:OFFSet?
```

Related Commands

DISPlay:SPECtrum[:MEASview<y>]:FREQuency:OFFSet[:SCALe] (on page 964)

Arguments

`<value>::<NRf>` specifies the frequency offset.

Range: [(center frequency) - (X scale) × 0.9] to [(center frequency) + (X scale) × 0.9]

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`DISPlay:SPECtrum:MEASview2:FREQuency:OFFSet 1.45GHz` sets the frequency offset to 1.45 GHz in the spectrum.

DISPlay:SPECtrum[:MEASview<y>]:FREQuency[:SCAle]

Sets or queries the horizontal range of the spectrum graph.

Conditions

Measurement views: Spectrum

Group

Display commands

Syntax

```
DISPlay:SPECtrum[:MEASview<y>]:FREQuency[:SCAle] <value>
DISPlay:SPECtrum[:MEASview<y>]:FREQuency[:SCAle]?
```

Related Commands

DISPlay:SPECtrum[:MEASview<y>]:FREQuency:OFFSet

Arguments

`<value>::<NRf>` specifies the horizontal range.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`DISPlay:SPECtrum:MEASview2:FREQuency:SCAle 10MHz` sets the horizontal range to 10 MHz.

DISPlay:SPECtrum[:MEASview<y>]:MARKer:NOISe:MODE

Determines whether to enable or disable the marker noise mode in the spectrum view. In this mode, the marker readout indicates amplitude in dBm/Hz. It is valid for all markers except for the reference marker.

NOTE: To use the marker noise mode, select dBm as the power unit by the `[SENSe]:POWer:UNITs` command.

NOTE: It is an overlapped command, which does not finish executing before the next command starts executing. Use the *OPC(?) command to synchronize all pending operations to the execution of this command.

Conditions

Measurement views: Spectrum

Group

Display commands

Syntax

```
DISPlay:SPECtrum[:MEASview<y>]:MARKer:NOISe:MODE { OFF | ON | 0 | 1 }
DISPlay:SPECtrum[:MEASview<y>]:MARKer:NOISe:MODE?
```

Related Commands

[SENSe]:POWer:UNITs (on page 2269)

Arguments

`OFF` or `0` disables the marker noise mode.

`ON` or `1` enables the marker noise mode.

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1.

Examples

`DISPlay:SPECtrum:MEASview2:MARKer:NOISe:MODE ON` enables the marker noise mode

DISPlay:SPECtrum[:MEASview<y>]:READout[:STATe]

Determines whether to show or hide the measurement parameter readouts in the Spectrum view.

Conditions

Measurement views: Spectrum

Group

Display commands

Syntax

```
DISPlay:SPECtrum[:MEASview<y>]:READout[:STATe] { OFF | ON | 0 | 1 }
DISPlay:SPECtrum[:MEASview<y>]:READout[:STATe]?
```

Arguments

`OFF` or `0` hides the parameter readouts.

`ON` or `1` shows the parameter readouts.

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1

Examples

`DISPLAY:SPECTRUM:MEASVIEW1:READOUT:STATE ON` shows the parameter readouts on the screen.

DISPlay:SPECtrum[:MEASview<y>][:SCALe]:LOG:STATe

Determines whether or not to set the horizontal axis logarithmic in the Spectrum view.

Conditions

Measurement views: Spectrum

Group

Display commands

Syntax

```
DISPlay:SPECtrum[:MEASview<y>][:SCALe]:LOG:STATe {OFF | ON | 0 | 1}
DISPlay:SPECtrum[:MEASview<y>][:SCALe]:LOG:STATe?
```

Related Commands

DISPlay:SPECtrum[:MEASview<y>]:X:LABel (on page 967)

Arguments

`OFF` or `0` sets the horizontal axis linear (default).

`ON` or `1` sets the horizontal axis logarithmic.

Executing `DISPlay:SPECtrum:SCALe:LOG:STATe ON` sets `DISPlay:SPECtrum:X:LABel SSFReq`.

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`DISPlay:SPECtrum:MEASview2:SCALe:LOG:STATe ON` sets the horizontal axis logarithmic in the Spectrum view.

DISPlay:SPECtrum[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions

Measurement views: Spectrum

Group

Display commands

Syntax

```
DISPlay:SPECtrum[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:SPECtrum[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or `0` hides the graticule grid.

`ON` or `1` shows the graticule grid.

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`DISPlay:SPECtrum:MEASview2:WINDow:TRACe:GRATicule:GRID:STATe ON` shows the graticule grid on the screen in the spectrum measurement.

DISPlay:SPECtrum[:MEASview<y>]:WINDow:TRACe:LEGend:STATe

Determines whether to show or hide the trace legend in the Spectrum view. The legend indicates the trace detection and function on the screen for each displayed spectrum trace.

Conditions

Measurement views: Spectrum

Group

Display commands

Syntax

```
DISPlay:SPECtrum[:MEASview<y>]:WINDow:TRACe:LEGend:STATe { OFF | ON | 0 | 1 }
DISPlay:SPECtrum[:MEASview<y>]:WINDow:TRACe:LEGend:STATe?
```

Arguments

`OFF` or `0` hides the trace legend.

`ON` or `1` shows the trace legend.

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`DISPlay:SPECtrum:MEASview2:WINDow:TRACe:LEGend:STATe ON` shows the trace legend on the screen in the spectrum measurement.

DISPlay:SPECtrum[:MEASview<y>]:X:LABel

Sets or queries the labels for the horizontal (X) axis in the Spectrum view. The labels are indicated beneath the spectrum graph on the screen.

Conditions

Measurement views: Spectrum

Group

Display commands

Syntax

```
DISPlay:SPECtrum[:MEASview<y>]:X:LABel { SSFReq | CFSPan }
DISPlay:SPECtrum[:MEASview<y>]:X:LABel?
```

Arguments

`SSFReq` sets the labels to the start and stop frequencies.

`CFSPan` sets the labels to the center frequency and span.

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`DISPlay:SPECtrum:MEASview2:X:LABel SSFReq` sets the labels to the start and stop frequencies for the horizontal axis in the Spectrum view.

DISPlay:SPECtrum[:MEASview<y>]:Y[:SCAle]

Sets or queries the vertical range of the spectrum graph.

Conditions

Measurement views: Spectrum

Group

Display commands

Syntax

```
DISPlay:SPECtrum[:MEASview<y>]:Y[:SCAle] <value>
DISPlay:SPECtrum[:MEASview<y>]:Y[:SCAle]?
```

Arguments

`<value>::<NRf>` specifies the vertical range. Range: 0.1 to 200 dB. The amplitude unit is set by the [SENSe]:POWer:UNITs (on page 2269) command.

You can omit the unit in the argument. When you include the unit in the argument, only dB is available. For the amplitude units Watts, Volts, and Amps, omit the unit in the argument.

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`DISPlay:SPECtrum:MEASview2:Y:SCAle 50dB` sets the vertical range to 50 dB in the Spectrum view.

DISPlay:SPECtrum[:MEASview<y>]:Y[:SCAle]:AUTO (No Query Form)

Rescales the vertical axis automatically to fit the waveform to the screen in the Spectrum view.

Conditions

Measurement views: Spectrum

Group

Display commands

Syntax

```
DISPlay:SPECtrum[:MEASview<y>]:Y[:SCAle]:AUTO
```

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`DISPlay:SPECtrum:MEASview2:Y:SCAle:AUTO` rescales the vertical scale automatically to fit the waveform to the screen

DISPlay:SPECtrum[:MEASview<y>]:Y[:SCAle]:OFFSET

Sets or queries the vertical offset (the value at the top edge of the vertical axis) of the spectrum graph.

Conditions

Measurement views: Spectrum

Group

Display commands

Syntax

```
DISPlay:SPECtrum[:MEASview<y>]:Y[:SCAle]:OFFSET <value>
DISPlay:SPECtrum[:MEASview<y>]:Y[:SCAle]:OFFSET?
```

Arguments

`<value>::<NRf>` specifies the vertical offset. Range: -270 to +150 dBm. The amplitude unit is set by the [SENSe]:POWer:UNITs (on page 2269) command (default: dBm).

You can omit the unit in the argument. When you include the unit in the argument, only dBm is allowed. For the other amplitude units, omit the unit in the argument.

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`DISPlay:SPECtrum:MEASview2:Y:SCAle:OFFSET -12.5` sets the vertical offset to -12.5 dBm

DISPlay:SPECtrum[:MEASview<y>]:Y[:SCAle]:PDIVision

Sets or queries the vertical scale (per division) of the spectrum graph.

Conditions

Measurement views: Spectrum

Group

Display commands

Syntax

```
DISPlay:SPECtrum[:MEASview<y>]:Y[:SCAle]:PDIVision <value>
DISPlay:SPECtrum[:MEASview<y>]:Y[:SCAle]:PDIVision?
```

Related Commands

[SENSe]:POWer:UNITs (on page 2269)

Arguments

<value>::<NRf> specifies the vertical scale (per division).

Range: 0.01 to 20 dB/div.

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

DISPlay:SPECtrum:MEASview2:Y:SCAle:PDIVision 0.5 sets the vertical scale to 0.5 dB/div.

DISPlay:SPECtrum[:MEASview<y>]:Y[:SCAle]:RESet (No Query Form)

Resets the vertical scale of the spectrum graph to the default values: Vertical offset = Reference level and Vertical scale = 100 dB

Conditions

Measurement views: Spectrum

Group

Display commands

Syntax

DISPlay:SPECtrum[:MEASview<y>]:Y[:SCAle]:RESet

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

DISPlay:SPECtrum:MEASview2:Y:SCAle:RESet resets the vertical scale to the default values in the Spectrum view

DISPlay:SPURious:MARKer:SHOW:STATe

Determines whether to show or hide the readout for the selected marker in the Spurious view.

Conditions

Measurement views: Spurious

Group

Display commands

Syntax

```
DISPlay:SPURious:MARKer:SHOW:STATe { OFF | ON | 0 | 1 }
DISPlay:SPURious:MARKer:SHOW:STATe?
```

Arguments

OFF or 0 hides the readout for the selected marker in the graph.

ON or 1 shows the readout for the selected marker in the graph.

Examples

DISPLAY:SPURIOUS:MARKER:SHOW:STATE ON shows the readout for the selected marker in the graph.

DISPlay:SPURious:RESet:SCALe (No Query Form)

Resets the horizontal and vertical scale to the default values described below in the Spurious view.

Vertical offset = 0 dBm, Vertical scale = 100 dB,

Horizontal offset = Center frequency, and Horizontal scale = Default span

Conditions

Measurement views: Spurious

Group

Display commands

Syntax

```
DISPlay:SPURious:RESet:SCALe
```

Arguments

None

Examples

DISPLAY:SPURIOUS:RESET:SCALE resets the horizontal and vertical scale to the default values.

DISPlay:SPURious:SCALe:LOG:STATe

Determines whether or not to set the horizontal axis logarithmic in the Spurious view.

Conditions

Measurement views: Spurious

Group

Display commands

Syntax

```
DISPlay:SPURious:SCALe:LOG:STATe { OFF | ON | 0 | 1 }
DISPlay:SPURious:SCALe:LOG:STATe?
```

Arguments

OFF or 0 sets the horizontal axis linear (default).

ON or 1 sets the horizontal axis logarithmic.

Examples

DISPLAY:SPURIOUS:SCALE:LOG:STATE ON sets the horizontal axis logarithmic in the Spurious view.

DISPlay:SPURious:SELect:NUMBer

Sets or queries the spurious number in the Spurious view.

Conditions

Measurement views: Spurious

Group

Display commands

Syntax

```
DISPlay:SPURious:SELect:NUMBer <number>
DISPlay:SPURious:SELect:NUMBer?
```

Arguments

`<number>::<NR1>` specifies the spurious number.

Range: 1 to the number of spurious signals.

Use the FETCh:SPURious:COUNt? (on page 1372) or READ:SPURious:COUNt? (on page 1827) query to get the number of spurious signals.

Examples

`DISPLAY:SPURIOUS:SELECT:NUMBER 7` selects the spurious #7.

DISPlay:SPURious:SHOW:LIMit

Sets or queries how to display the limits.

Conditions

Measurement views: Spurious

Group

Display commands

Syntax

```
DISPlay:SPURious:SHOW:LIMit { SHADed | LINE | OFF }
DISPlay:SPURious:SHOW:LIMit?
```

Arguments

`SHADed` displays the limits with shade.

`LINE` displays the limits with line only.

`OFF` hides the limits.

Examples

`DISPLAY:SPURIOUS:SHOW:LIMIT LINE` displays the limits with line only.

DISPlay:SPURious:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions

Measurement views: Spurious

Group

Display commands

Syntax

```
DISPlay:SPURious:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:SPURious:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or `0` hides the graticule grid.

`ON` or `1` shows the graticule grid.

Examples

`DISPLAY:SPURIOUS:WINDOW:TRACE:GRATICULE:GRID:STATE ON` shows the graticule grid on the screen.

DISPlay:SPURious:X[:SCALe]:AUTO (No Query Form)

Rescales the horizontal axis automatically to fit the waveform to the screen in the Spurious view.

Conditions

Measurement views: Spurious

Group

Display commands

Syntax

```
DISPlay:SPURious:X[:SCALe]:AUTO
```

Arguments

None

Examples

`DISPLAY:SPURIOUS:X:SCALE:AUTO` rescales the horizontal scale automatically to fit the waveform to the screen.

DISPlay:SPURious:X[:SCALe]:STARt

Sets or queries the minimum horizontal value (left edge) of the spectrum graph in the Spurious view.

Conditions

Measurement views: Spurious

Group

Display commands

Syntax

```
DISPlay:SPURious:X[:SCALe]:STARt <value>
DISPlay:SPURious:X[:SCALe]:STARt?
```

Arguments

<value>::<NRf> specifies the minimum horizontal value.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples

DISPLAY:SPURIOUS:X:SCALE:START 1.61GHz sets the minimum horizontal value to 1.61 GHz in the spectrum graph.

DISPlay:SPURious:X[:SCALe]:STOP

Sets or queries the maximum horizontal value (right edge) of the spectrum graph in the Spurious view.

Conditions

Measurement views: Spurious

Group

Display commands

Syntax

```
DISPlay:SPURious:X[:SCALe]:STOP <value>
DISPlay:SPURious:X[:SCALe]:STOP?
```

Arguments

<value>::<NRf> specifies the minimum horizontal value.

Range:

- RSA5103AB – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples

DISPLAY:SPURIOUS:X:SCALE:STOP 2.16GHz sets the maximum horizontal value to 2.16 GHz in the spectrum graph.

DISPlay:SPURious:Y[:SCALe]

Sets or queries the vertical range of the spectrum graph in the Spurious view.

Conditions

Measurement views: Spurious

Group

Display commands

Syntax

```
DISPlay:SPURious:Y[:SCALe] <value>
DISPlay:SPURious:Y[:SCALe]?
```

Arguments

<value>::<NRf> specifies the vertical range. Range: 0.1 to 200 dB. The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples

DISPLAY:SPURIOUS:Y:SCALE 100 sets the vertical range to 100 dB in the Spurious view.

DISPlay:SPURious:Y[:SCALe]:AUTO (No Query Form)

Rescales the vertical axis automatically to fit the waveform to the screen in the Spurious view.

Conditions

Measurement views: Spurious

Group

Display commands

Syntax

DISPlay:SPURious:Y[:SCALe]:AUTO

Arguments

None

Examples

DISPLAY:SPURIOUS:Y:SCALE:AUTO rescales the vertical scale automatically to fit the waveform to the screen.

DISPlay:SPURious:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (the value at the top edge of the vertical axis) of the spectrum graph in the Spurious view.

Conditions

Measurement views: Spurious

Group

Display commands

Syntax

DISPlay:SPURious:Y[:SCALe]:OFFSet <value>
DISPlay:SPURious:Y[:SCALe]:OFFSet?

Arguments

<value>::<NRf> specifies the vertical offset. Range: -270 to +150 dBm.The unit can be changed by the [SENSe]:POWer:UNITs (on page 2269) command.

Examples

DISPLAY:SPURIOUS:Y:SCALE:OFFSET -12.5 sets the vertical offset to -12.5 dBm in the spectrum graph.

DISPlay:TDIagram[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen in the Trellis diagram.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Trellis diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Trellis diagram).

Group

Display commands

Syntax

```
DISPlay:TDIagram[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:TDIagram[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or 0 hides the graticule grid.

`ON` or 1 shows the graticule grid.

Returns

OFF or 0

ON or 1

Examples

`DISPLay:TDIagram:MEASview2:WINDOW:TRACe:GRATICULE:GRID:STATe ON` shows the graticule grid on the screen in the Trellis diagram.

DISPlay:TDIagram[:MEASview<y>]:Y[:SCALe]

Sets or queries the vertical range of the trellis diagram.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Trellis diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Trellis diagram).

Group

Display commands

Syntax

```
DISPlay:TDIagram[:MEASview<y>]:Y[:SCALe] <value>
DISPlay:TDIagram[:MEASview<y>]:Y[:SCALe]?
```

Arguments

`<value>::<NRf>` specifies the vertical range.

Range: 1 ° to 1 T°.

Returns

<NRf>

Examples

`DISPlay:TDIagram:MEASview2:Y:SCALe 2.5` sets the vertical range to 2.5° in the Trellis diagram.

DISPlay:TDIagram[:MEASview<y>]:Y[:SCALe]:AUTO (No query form)

Sets the vertical scale automatically to fit the waveform to the screen in the Trellis diagram.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Trellis diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Trellis diagram).

Group

Display commands

Syntax

`DISPlay:TDIagram[:MEASview<y>]:Y[:SCALe]:AUTO`

Arguments

None

Examples

`DISPlay:TDIagram:MEASview2:Y:SCALe:AUTO` sets the vertical scale automatically to fit the waveform to the screen.

DISPlay:TDIagram[:MEASview<y>]:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (center point of the vertical axis) of the Trellis diagram.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Trellis diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Trellis diagram).

Group

Display commands

Syntax

```
DISPlay:TDIagram[:MEASview<y>]:Y[:SCALe]:OFFSet <value>
DISPlay:TDIagram[:MEASview<y>]:Y[:SCALe]:OFFSet?
```

Arguments

`<value>::<NRf>` specifies the vertical offset.

Range: -1 T° to +1 T°.

Returns

<NRf>

Examples

`DISPlay:TDIagram:MEASview2:Y:SCALe:OFFSet -28.5` sets the vertical offset to -28.5 ° in the Trellis diagram measview 2.

DISPlay:TG:MEASview:DELete TXGain (No Query Form)

This command removes the Transmission Gain display icon from the Selected displays filed of the Selected Displays window.

Conditions

Measurement view: Select Displays window

Group

Display commands

Syntax

```
DISPlay:TG:MEASview:DELete TXGain
```

Arguments

Examples

```
DISPLAY:TG:MEASVIEW:DELETE TXGAIN
```

DISPlay:TG:MEASview:NEW TXGain (No Query Form)

This command opens a new Transmission Gain measurement display window.

Conditions

Measurement view: Select Displays window

Group

Display commands

Syntax

```
DISPlay:TG:MEASview:NEW TXGain
```

DISPlay:TG:MEASview:SELect TXGain

This command selects the Transmission Gain display icon in the Select Display window and queries if the display is selected or not.

Conditions

Measurement view: Select Displays window

Group

Display commands

Syntax

```
DISPlay:TG:MEASview:SELect TXGain { OFF | ON | 0 | 1 }
DISPlay:TG:MEASview:SELect TXGain?
```

Arguments

`OFF` or `0` deselects the Transmission Gain display icon.

`ON` or `1` selects the Transmission Gain display icon.

Returns

0 means the icon is not selected.

1 means the icon is selected.

Examples

```
DISPLAY:TG:MEASVIEW:SELECT TXGAIN
```

DISPlay:TOVerview[:MEASview<y>]:WINDow:NAVigator:STATe

Sets or queries whether the navigator view of the Time Overview display is on or off.

Conditions

Measurement view: Time Overview with specified Measurement view number

Group

Display commands

Syntax

```
DISPlay:TOVerview[:MEASview<y>]:WINDow:NAVigator:STATe { OFF | ON | 0 | 1 }
DISPlay:TOVerview[:MEASview<y>]:WINDow:NAVigator:STATe?
```

Arguments

`OFF` or 0 sets the Time Overview display to not show in Navigator View mode.

`ON` or 1 sets the Time Overview display to show in Navigator View mode.

Returns

OFF or 0 means that the Time Overview display is not in Navigator View mode.

ON or 1 means that the Time Overview display is in Navigator View mode.

Examples

`DISPLAY:TOVERVIEW:MEASVIEW1:WINDOW:NAVIGATOR:STATE ON` will set the Time Overview display to show in Navigator View mode.

DISPlay:TOVerview[:MEASview<y>]:WINDow:TIME:MODE

Sets or queries the type of time analysis to be performed for the Time Overview display.

Conditions

Measurement view: Time Overview with specified Measurement view number

Group

Display commands

Syntax

```
DISPlay:TOVerview[:MEASview<y>]:WINDow:TIME:MODE { ANALysis | SPECtrum | LINKed }
DISPlay:TOVerview[:MEASview<y>]:WINDow:TIME:MODE?
```

Related Commands

Arguments

`ANALysis` sets the type of analysis time to Analysis length, which is the period of time within the acquisition record over which all other measurements (such as Amplitude vs. Time) are made.

`SPECtrum` sets the type of analysis time to Spectrum length, which is the period of time within the acquisition record over which the spectrum is calculated.

`LINKed` sets the type of analysis time to lock the Spectrum length and Analysis length together so that the data used to produce the Spectrum display is also used for measurement displays.

Returns

`ANAL` means that the type of time analysis is set to Analysis.

`SPEC` means that the type of time analysis is set to Spectrum.

`LINK` means that the type of time analysis is set to Linked.

Examples

`DISPLAY:TOVERVIEW:WINDOW:TIME:MODE?` might return `SPEC`, which indicates that the type of analysis time is set to Spectrum, which is the period of time within the acquisition record over which the spectrum is calculated.

DISPlay:TOVerview[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions

Measurement views: Time overview

Group

Display commands

Syntax

```
DISPlay:TOVerview[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:TOVerview[:MEASview<y>]:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or `0` hides the graticule grid.

ON or 1 shows the graticule grid.

Examples

DISPLAY:TOVERVIEW:MEASVIEW1:WINDOW:TRACE:GRATICULE:GRID:STATE ON shows the graticule grid on the screen in the time overview Measurement View 1.

DISPlay:TOVerview[:MEASview<y>]:WINDow:TRACe:LEGend:STATe

Sets or queries the trace legend state in the Time Overview display to on (showing) or off (hidden).

Conditions

Measurement view: Time Overview with specified Measurement view number

Group

Display commands

Syntax

```
DISPlay:TOVerview[:MEASview<y>]:WINDow:TRACe:LEGend:STATe { OFF | ON | 0 | 1 }
DISPlay:TOVerview[:MEASview<y>]:WINDow:TRACe:LEGend:STATe?
```

Related Commands

Arguments

OFF or 0 hides the trace legend in the Time Overview display.

ON or 1 shows the trace legend in the Time Overview display.

Returns

0 means the trace legend in the Time Overview display is hidden (off).

1 means the trace legend in the Time Overview display is showing (on).

Examples

DISPLAY:TOVERVIEW:MEASVIEW1:WINDOW:TRACE:LEGEND:STATE ON will turn on the trace legend in the Time Overview display.

DISPLAY:TOVERVIEW:MEASVIEW1:WINDOW:TRACE:LEGEND:STATE? might return 1, which indicates that the trace legend in the Time Overview display is showing (on).

DISPlay:TOVerview[:MEASview<y>]:X[:SCALe]

Sets or queries the horizontal scale (full-scale time) of the time overview.

Conditions

Measurement views: Time overview with specified Measurement view number

Group

Display commands

Syntax

```
DISPlay:TOVerview[:MEASview<y>]:X[:SCALe] <value>
DISPlay:TOVerview[:MEASview<y>]:X[:SCALe]?
```

Related Commands

DISPlay:TOVerview[:MEASview<y>]:X[:SCALe]:OFFSet (on page 982)

Arguments

<value>::<NRf> specifies the horizontal scale in full-scale time.

Range: 10 ns to the acquisition memory capacity.

You can see the acquisition memory capacity using the [SENSe]:ACQuisition:MEMory:CAPacity[:TIME]? (on page 1944) query.

Examples

DISPLAY:TOVERVIEW:MEASVIEW1:X:SCALE 12.5us sets the horizontal scale to 12.5 µs.

DISPlay:TOVerview[:MEASview<y>]:X[:SCALe]:AUTO (No Query Form)

Sets the horizontal scale and offset automatically to fit the waveform to the screen in the time overview.

Conditions

Measurement views: Time overview with specified Measurement view number

Group

Display commands

Syntax

DISPlay:TOVerview[:MEASview<y>]:X[:SCALe]:AUTO

Arguments

None

Examples

DISPLAY:TOVERVIEW:MEASVIEW1:X:SCALE:AUTO sets the horizontal scale and offset automatically to fit the waveform on the trace of Time overview Measurement View 1.

DISPlay:TOVerview[:MEASview<y>]:X[:SCALe]:OFFSet

Sets or queries the minimum horizontal value (left edge) of the time overview.

Conditions

Measurement views: Time overview with specified Measurement view number

Group

Display commands

Syntax

```
DISPlay:TOVerview[:MEASview<y>]:X[:SCALe]:OFFSet <value>
DISPlay:TOVerview[:MEASview<y>]:X[:SCALe]:OFFSet?
```

Related Commands

DISPlay:TOVerview[:MEASview<y>]:X[:SCALe] (on page 981)

Arguments

<value>::<NRf> specifies the minimum horizontal value.

Range: [(analysis offset) - (X scale) × 0.9] to [(analysis offset) + (analysis length) - (X scale) × 0.1]

Examples

DISPLAY:TOVERVIEW:MEASVIEW1:X:SCALE:OFFSET 800ns sets the minimum horizontal value to 800 ns in the Time overview Measurement View 1.

DISPlay:TOVerview[:MEASview<y>]:Y[:SCALe]

Sets or queries the vertical range of the time overview.

Conditions

Measurement views: Time overview with specified Measurement view number

Group

Display commands

Syntax

```
DISPlay:TOVerview[:MEASview<y>]:Y[:SCALe] <value>
DISPlay:TOVerview[:MEASview<y>]:Y[:SCALe]?
```

Related Commands

DISPlay:TOVerview[:MEASview<y>]:Y[:SCALe]:OFFSet (on page 984)

Arguments

<value>::<NRf> specifies the vertical range. Range: 0.1 to 200 dB.

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf>

Examples

DISPLAY:TOVERVIEW:MEASVIEW1:Y:SCALE 50 sets the vertical range to 50 dBm in the time overview.

DISPlay:TOVerview[:MEASview<y>]:Y[:SCALe]:AUTO (No Query Form)

Sets the vertical scale and offset automatically to fit the waveform to the screen in the time overview.

Conditions

Measurement views: Time overview with specified Measurement view number

Group

Display commands

Syntax

```
DISPlay:TOVerview[:MEASview<y>]:Y[:SCALe]:AUTO
```

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Examples

`DISPLAY:TOVERVIEW:MEASVIEW1:Y:SCALE:AUTO` sets the vertical scale and offset automatically to fit the waveform to the screen.

DISPlay:TOVerview[:MEASview<y>]:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (the value at the top edge of the vertical axis) in the time overview.

Conditions

Measurement views: Time overview with specified Measurement view number

Group

Display commands

Syntax

```
DISPlay:TOVerview[:MEASview<y>]:Y[:SCALe]:OFFSet <value>
DISPlay:TOVerview[:MEASview<y>]:Y[:SCALe]:OFFSet?
```

Related Commands

DISPlay:TOVerview[:MEASview<y>]:Y[:SCALe] (on page 983)

Arguments

`<value>::<NRf>` specifies the vertical offset. Range: -170 to +50 dBm.

Returns

`<NRf>`

Examples

`DISPLAY:TOVERVIEW:MEASVIEW1:Y:SCALE:OFFSET -80` sets the vertical offset to -80 dBm in the time overview Measurement View 1.

DISPlay:TOVerview[:MEASview<y>]:Y[:SCALe]:RESCale (No Query Form)

Sets the vertical scale automatically to fit the waveform to the screen in the time overview.

Conditions

Measurement views: Time overview with specified Measurement view number

Group

Display commands

Syntax

```
DISPlay:TOVerview[:MEASview<y>]:Y[:SCALe]:RESCale
```

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Examples

`DISPLAY:TOVERVIEW:MEASVIEW1:Y:SCALE:RESCALE` sets the vertical scale automatically to fit the waveform to the screen.

DISPlay:TXGain:MARKer:SHOW:STATe

Displays or hides the marker readout, but not the marker itself, on the graph area.

Conditions

Measurement view: Transmission Gain

Group

Display commands

Syntax

```
DISPlay:TXGain:MARKer:SHOW:STATe { OFF | ON | 0 | 1 }
DISPlay:TXGain:MARKer:SHOW:STATe?
```

Arguments

`OFF` or `0` hides the readout for the selected marker.

`ON` or `1` shows the readout for the selected marker.

Returns

OFF or 0 means that the readout for the selected marker is hidden.

ON or 1 means that the readout for the selected marker is showing.

Examples

`DISPLAY:TXGAIN:MARKER:SHOW:STATE ON` shows the readout for the marker on the display.

DISPlay:TXGain[:SCALe]:AUTO (No Query Form)

Automatically rescales the horizontal and vertical scale.

Conditions

Measurement view: Transmission Gain

Group

Display commands

Syntax

```
DISPlay:TXGain[:SCALe]:AUTO
```

Arguments

None

Examples

`DISPLAY:TXGAIN[:SCALE]:AUTO` rescales the horizontal and vertical scale automatically.

DISPlay:TXGain:WINDow:TRACe:GRATicule:GRID:STATe

Displays or hides the graticule grid in the display. Queries whether the graticule grid is hidden or showing.

Conditions

Measurement view: Transmission Gain

Group

Display commands

Syntax

```
DISPlay:TXGain:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:TXGain:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or 0 hides the graticule.

`ON` or 1 shows the graticule.

Returns

`OFF` or 0 means that the graticule is hidden.

`ON` or 1 means that the graticule is showing.

Examples

`DISPLAY:TXGAIN:WINDOW:TRACE:GRATICULE:GRID:STATE OFF` will hide the graticule.

DISPlay:TXGain:WINDow:TRACe:POINts:SHOW:STATe

Sets to mark each measurement point on the trace.

Conditions

Measurement view: Transmission Gain

Group

Display commands

Syntax

```
DISPlay:TXGain:WINDow:TRACe:POINts:SHOW:STATe { OFF | ON | 0 | 1 }
DISPlay:TXGain:WINDow:TRACe:POINts:SHOW:STATe?
```

Arguments

`OFF` or 0 hides each measurement point on the trace.

`ON` or 1 shows each measurement point on the trace.

Returns

`OFF` or 0 means that each measurement point on the trace is hidden.

`ON` or 1 means that each measurement point on the trace is showing.

Examples

`DISPLAY:TXGAIN:WINDOW:TRACE:POINTS:SHOW:STATE 1` shows each measurement point on the trace.

DISPlay:TXGain:X[:SCALe]:AUTO (No Query Form)

Resets the scale of the horizontal axis to contain the complete trace.

Conditions

Measurement view: Transmission Gain

Group

Display commands

Syntax

```
DISPlay:TXGain:X[:SCALe]:AUTO
```

Arguments

None

Examples

`DISPLAY:TXGAIN:X:SCALE:AUTO` resets the scale of the horizontal axis to contain the complete trace.

DISPlay:TXGain:X[:SCALe]:LOG:STATe

Sets or queries the display to show the frequency axis in a logarithmic scale.

Conditions

Measurement view: Transmission Gain

Group

Display commands

Syntax

```
DISPlay:TXGain:X[:SCALe]:LOG:STATe { OFF | ON | 0 | 1 }
DISPlay:TXGain:X[:SCALe]:LOG:STATe?
```

Arguments

`OFF` or `0` sets the frequency axis in linear scale.

`ON` or `1` sets the frequency axis in logarithmic scale.

Returns

OFF or 0 means that the frequency axis is set to linear scale.

ON or 1 means that the frequency axis is set to logarithmic scale.

Examples

`DISPLAY:TXGAIN:X:SCALE:LOG:STATE 1` sets the frequency axis to logarithmic scale.

DISPlay:TXGain:X[:SCALe]:STARt

Sets the Zoom Start frequency. This only affects the start frequency shown on the graph.

Conditions

Measurement view: Transmission Gain

Group

Display commands

Syntax

```
DISPlay:TXGain:X[:SCALe]:STARt <value>
DISPlay:TXGain:X[:SCALe]:STARt?
```

Arguments

<value>::=<NRf> is a floating point number that is the zoom start frequency.

Returns

See Arguments.

Examples

DISPLAY:TXGAIN:X:SCALE:START 1E9 sets the zoom start frequency to 1 GHz.

DISPlay:TXGain:X[:SCALe]:STOP

Sets the Zoom Stop frequency. This only affects the stop frequency shown on the graph.

Conditions

Measurement view: Transmission Gain

Group

Display commands

Syntax

```
DISPlay:TXGain:X[:SCALe]:STOP <value>
DISPlay:TXGain:X[:SCALe]:STOP?
```

Arguments

<value>::=<NRf> specifies the zoom stop frequency.

Returns

A floating point value that is the zoom stop frequency.

Examples

DISPLAY:TXGAIN:X:SCALE:STOP 1E9 sets the zoom stop frequency to 1 GHz.

DISPlay:TXGain:Y[:SCALe]

Sets or queries the vertical scale value in dB.

Conditions

Measurement view: Transmission Gain

Group

Display commands

Syntax

```
DISPlay:TXGain:Y[:SCALe] <value>
DISPlay:TXGain:Y[:SCALe]?
```

Arguments

<value>::=<NRf> specifies the vertical scale in dB.

Returns

A floating point value that is the vertical scale in dB.

Examples

DISPLAY:TXGAIN:Y:SCALE 100 sets the vertical scale to 100 dB.

DISPlay:TXGain:Y[:SCALe]:AUTO (No Query Form)

Automatically scales the plot vertically.

Conditions

Measurement view: Transmission Gain

Group

Display commands

Syntax

```
DISPlay:TXGain:Y[:SCALe]:AUTO
```

Arguments

None

Examples

DISPLAY:TXGAIN:Y:SCALE:AUTO automatically scales the plot vertically.

DISPlay:TXGain:Y[:SCALe]:BOTTom

Sets or queries the bottom of the vertical position of the display.

Conditions

Measurement view: Transmission Gain

Group

Display commands

Syntax

```
DISPlay:TXGain:Y[:SCALe]:BOTTom <value>
DISPlay:TXGain:Y[:SCALe]:BOTTom?
```

Arguments

<value>::=<NRf> specifies the bottom of the vertical position of the display.

Returns

A floating point value that is the bottom of the vertical position of the display.

Examples

`DISPLAY:TXGAIN:Y:SCALE:BOTTOM -100` sets the bottom of Y axis to -100 dB.

DISPlay:TXGain:Y[:SCALe]:PDIVision

Sets or queries the vertical scale (per division) of the graph.

Conditions

Measurement view: Transmission Gain

Group

Display commands

Syntax

```
DISPlay:TXGain:Y[:SCALe]:PDIVision <value>
DISPlay:TXGain:Y[:SCALe]:PDIVision?
```

Arguments

`<value>::=<NRf>` specifies the vertical scale (per division).

Returns

A floating point value that is the vertical scale (per division) of the graph.

Examples

`DISPLAY:TXGAIN:Y:SCALE:PDIVision 0.5` sets the vertical scale to 0.5 dB/div.

DISPlay:TXGain:Y[:SCALe]:POSition

Sets or queries the vertical position of the trace.

Conditions

Measurement view: Transmission Gain

Group

Display commands

Syntax

```
DISPlay:TXGain:Y[:SCALe]:POSition <value>
DISPlay:TXGain:Y[:SCALe]:POSition?
```

Arguments

`<value>::=<NRf>` specifies the vertical position.

Returns

A floating point value that is the vertical position.

Examples

`DISPLAY:TXGAIN:Y:SCALE:PDIVision 0` sets the vertical position to 0 dB.

DISPlay:TXGain:Y[:SCALe]:TOP

Sets or queries the top of the vertical scale.

Conditions

Measurement view: Transmission Gain

Group

Display commands

Syntax

```
DISPlay:TXGain:Y[:SCALe]:TOP <value>
DISPlay:TXGain:Y[:SCALe]:TOP?
```

Arguments

`<value>::=<NRf>` specifies the top of the vertical scale.

Returns

A floating point value that is the top of the vertical scale.

Examples

`DISPLAY:TXGAIN:Y:SCALE:TOP 10` sets the vertical top to 10 dB.

:DISPlay:WINDow:ACQuisition:BANdwidth:SHOw:STAte {set and query}

Sets or queries the state of show acquisition bandwidth checkbox.

Conditions

Measurement views: Spectrum, Spectrogram

Group

Display commands

Syntax

```
:DISPlay:WINDow:ACQuisition:BANdwidth:SHOw:STAte {0|1|TRUE |FALSE}
:DISPlay:WINDow:ACQuisition:BANdwidth:SHOw:STAte?
```

Arguments

0 is used to change 'Show Acquisition Bandwidth' checkbox state from checked to unchecked.

1 is used to change 'Show Acquisition Bandwidth' checkbox state from unchecked to checked.

Examples

`:DISPlay:WINDow:ACQuisition:BANdwidth:SHOw:STAte 1` changes the state of 'Show Acquisition Bandwidth' checkox to checked.

DISPlay:WINDow:ACTive:MEASurement? (Query Only)

Queries the active measurement views.

Conditions

Measurement views: All

Group

Display commands

Syntax

```
DISPlay:WINDow:ACTive:MEASurement?
```

Arguments

None

Returns

<view1>,<view2>,...,<view(n)>

Where

`<view(n)>::<string>` is the view name as shown in the following table.

Measurement view mnemonic

Return value	Measurement view	Display group
"SPEC"	Spectrum	General signal viewing
"DPX"	DPX (Digital Phosphor) spectrum	
"MAGVT"	Amplitude versus Time	
"ACP"	Channel power and ACPR	
"FVT"	Frequency versus Time	
"PHVT"	Phase versus Time	
"IQVT"	RF I&Q versus Time	
"SGRam"	Spectrogram	
"TOV"	Time overview	
"AM"	Amplitude modulation	Analog Modulation
"FM"	Frequency modulation	
"PM"	Phase modulation	

Return value	Measurement view	Display group
"CONS"	Constellation	General purpose digital modulation
"DIQV"	Demodulated I&Q versus Time	
"EDI"	Eye Diagram	
"EVM"	EVM versus Time	
"FVDT"	Frequency deviation versus Time	
"MERR"	Magnitude error versus Time	
"PERR"	Phase error versus Time	
"SIGN"	Signal quality	
"STAB"	Symbol table	
"TDI"	Trellis Diagram	
"CCDF"	CCDF	RF measurements
"MCP"	MCPR (Multiple Carrier Power Ratio)	
"OBW"	Occupied bandwidth	
"PNO"	Phase noise	
"SPUR"	Spurious	
"STAT"	Pulse statistics	Pulsed RF
""RES"	Pulse table (results table)	
"TRAC"	Pulse trace	

DISPlay:WINDow:APPLy:SOURce:COLOr {set and query}

Sets or queries the state of apply source color to trace checkbox.

Conditions

Measurement views: All

Applicable for only MSO/LPD/SX.

Group

Display commands

Syntax

```
:DISPlay:WINDow:APPLy:SOURce:COLOr {0|1|TRUE |FALSE}
```

```
:DISPlay:WINDow:APPLy:SOURce:COLOr?
```

Argument

`0` is used to change 'Apply Source Color to Trace' checkbox state from unchecked to checked.

`1` is used to change 'Apply Source Color to Trace' checkbox state from checked to unchecked.

Examples

`:DISPlay:WINDow:APPLy:SOURce:COLOr 1` changes the state of 'Apply Source Color to Trace' checkox to checked.

DISPlay:WINDow:COLor:SCHeme

Sets or queries the color scheme for displaying traces and background on the screen.

Conditions

Measurement views: All

Group

Display commands

Syntax

```
DISPlay:WINDow:COLor:SCHeme { THUNderstorm | BLIZzard | CLASsic }
DISPlay:WINDow:COLor:SCHeme?
```

Arguments

`THUNderstorm` displays the backgound in dark blue.

`BLIZzard` displays the backgound in white. It saves ink when printing the screen image.

`CLASsic` displays the backgound in black (default).

Examples

`DISPLAY:WINDOW:COLOR:SCHEME BLIZzard` displays the backgound in white.

DISPlay:WINDow:OPTimized:MEASurement? (Query Only)

Queries the measurement views that are optimized. "Optimized" means that there is a perfect match between the view's settings and the actual acquisition parameters to meet the specifications. When multiple measurements are running at one time, the measurements can have different requirements for setting the acquisition hardware. You can make a measurement optimized by selecting it using the following commands:

- DISPlay:GENeral:MEASview<y>:SELect (on page 757) for the general signal viewing
- DISPlay:DDEMod[:MEASview<y>]:SELect (on page 710) for the digital modulation views
- DISPlay:GPRF:MEASview:SELect (on page 760) for the RF measurement views
- DISPlay:PULSe:MEASview<y>:SELect (on page 910) for the pulsed RF measurement views

Conditions

Measurement views: All

Group

Display commands

Syntax

```
DISPlay:WINDow:OPTimized:MEASurement?
```

Arguments

None

Returns

```
<view1>,<view2>,...,<view(n)>
```

Where

`<view(n)>::<string>` is the view name as shown in the table. The following table shows the view arguments. The arguments are the string type.

General signal viewing views

Argument	View
SPECtrum	Spectrum
DPX	DPX (Digital Phosphor) spectrum
AVTime	Amplitude versus Time
FVTime	Frequency versus Time
PHVTime	Phase versus Time
IQVTime	IQ versus Time
SGRam	Spectrogram
TOVerview	Time overview

Examples

`DISPLAY:WINDOW:OPTIMIZED:MEASUREMENT?` might return `"SPEC","MCP"`, indicating that the views of spectrum and MCPR are optimized.

DISPlay:WLAN:AD:CONSte:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticules in the 802.11ad Constellation display.

Conditions

Measurement view: 802.11ad Constellation

Group

Display commands

Syntax

```
DISPlay:WLAN:AD:CONSte:WINDow:TRACe:GRATicule:GRID:STATe { 0 | 1 | ON | OFF }
DISPlay:WLAN:AD:CONSte:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

0 or OFF hides the graticules in the graph

1 or ON shows the graticules in the graph

Returns

0 or OFF means that the graticules in the graph are hidden

1 or ON means that the graticules in the graph are showing

Examples

`DISPLAY:WLAN:AD:CONSTE:WINDOW:TRACE:GRATICULE:GRID:STATE 1` shows the graticules in the display.

DISPlay:WLAN:AD:CONSte:WINDow:TRACe:MARKer:STATe

Determines whether to show or hide marker readout in graph in the 802.11ad Constellation display.

Conditions

Measurement view: 802.11ad Constellation

Group

Display commands

Syntax

```
DISPlay:WLAN:AD:CONSte:WINDow:TRACe:MARKer:STATe { 0 | 1 | ON | OFF }
DISPlay:WLAN:AD:CONSte:WINDow:TRACe:MARKer:STATe?
```

Arguments

`OFF` or `0` hides the marker readout in the graph

`ON` or `1` shows the marker readout in the graph

Returns

`0` or `OFF` means that the graticules in the graph are hidden

`1` or `ON` means that the graticules in the graph are shown

Examples

`DISPLAY:WLAN:AD:CONSTE:WINDOW:TRACE:MARKER:STATE 1` shows the marker readout in the display.

DISPlay:WLAN:AD:CONSte:ZOOM:SCALe

Sets or queries the zoom scale value in the Scale tab of the Settings control panel for the 802.11ad Constellation display.

Conditions

Measurement view: 802.11ad Constellation

Group

Display commands

Syntax

```
DISPlay:WLAN:AD:CONSte:ZOOM:SCALe <value>
DISPlay:WLAN:AD:CONSte:ZOOM:SCALe?
```

Arguments

<value>=<NRf> where range is 0.1 to 10

Returns

<NRf> where range is 0.1 to 10

Examples

DISPLAY:WLAN:AD:CONSTE:ZOOM:SCALE 5 sets the zoom scale value to 5.

DISPlay:WLAN:AD:CONSte:ZOOM[:SCALe]:X:OFFSet

Sets or queries the Horizontal Position value in the Scale tab of the Settings control panel for the 802.11ad Constellation display.

Conditions

Measurement view: 802.11ad Constellation

Group

Display commands

Syntax

```
DISPlay:WLAN:AD:CONSte:ZOOM[:SCALe]:X:OFFSet <value>
DISPlay:WLAN:AD:CONSte:ZOOM[:SCALe]:X:OFFSet?
```

Arguments

<value>=<NRf> where range is -5.1 to 5.1

Returns

<NRf> where range is -5.1 to 5.1

Examples

DISPLAY:WLAN:AD:CONSTE:ZOOM[:SCALE]:X:OFFSET 3.5 sets the Horizontal Position to 3.5.

DISPlay:WLAN:AD:CONSte:ZOOM[:SCALe]:Y:OFFSet

Sets or queries the Vertical Position value in the Scale tab of the Settings control panel for the 802.11ad Constellation display.

Conditions

Measurement view: 802.11ad Constellation

Group

Display commands

Syntax

```
DISPlay:WLAN:AD:CONSte:ZOOM[:SCALe]:Y:OFFSet <value>
DISPlay:WLAN:AD:CONSte:ZOOM[:SCALe]:Y:OFFSet?
```

Arguments

<value>=<NRf> where range is -5.1 to 5.1

Returns

<NRf> where range is -5.1 to 5.1

Examples

DISPLAY:WLAN:AD:CONSTE:ZOOM[:SCALE]:Y:OFFSET -3.5 sets the Vertical Position to -3.5.

DISPlay:WLAN:AD:DATA:RADix

Set or queries the symbol radix type (set in the Prefs tab of the Settings control panel) in the display.

Conditions

Measurement views: All 802.11ad Symbol Table and Constellation

Group

Display commands

Syntax

```
DISPlay:WLAN:AD:DATA:RADix {DEC,BIN}
DISPlay:WLAN:AD:DATA:RADix?
```

Arguments

DEC sets the radix type to Decimal

BIN sets the radix type to Binary

Returns

See Arguments

Examples

DISPLAY:WLAN:AD:DATA:RADIX might return BIN, indicating the data radix selected is binary.

DISPlay:WLAN:AD:EVM:SCALe:AUTO (No Query Form)

Rescales both horizontal and vertical axis automatically to fit the waveform to the screen in the display.

Conditions

Measurement views: 802.11ad EVM vs Time

Group

Display commands

Syntax

```
DISPlay:WLAN:AD:EVM:SCALe:AUTO
```

Arguments

None

Returns

None

Examples

`DISPLAY:WLAN:AD:EVM:SCALE:AUTO` fits the waveform to the graph both horizontally and vertically.

DISPlay:WLAN:AD:EVM:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule in the display.

Conditions

Measurement views: 802.11ad EVM vs Time

Group

Display commands

Syntax

```
DISPlay:WLAN:AD:EVM:WINDow:TRACe:GRATicule:GRID:STATe { 0 | 1 | ON | OFF }
DISPlay:WLAN:AD:EVM:WINDow:TRACe:GRATicule:GRID:STATe ?
```

Arguments

`OFF` or `0` hides the graticules in the graph

`1` or `ON` shows the graticules in the graph

Returns

0 or OFF means that the graticules in the graph are hidden

1 or ON means that the graticules in the graph are showing

Examples

`DISPLAY:WLAN:AD:EVM:WINDOW:TRACE:GRATICULE:GRID:STATE OFF` turns off the graticule in the graph.

DISPlay:WLAN:AD:EVM:WINDow:TRACe:MARKer:STATe

Determines whether to show or hide the readout for the selected marker in the display.

Conditions

Measurement views: 802.11ad EVM vs Time

Group

Display commands

Syntax

```
DISPlay:WLAN:AD:EVM:WINDow:TRACe:MARKer:STATe { 0 | 1 | ON | OFF }
DISPlay:WLAN:AD:EVM:WINDow:TRACe:MARKer:STATe ?
```

Arguments

`OFF` or `0` hides the marker readout in the graph

`1` or `ON` shows the marker readout in the graph

Returns

0 or OFF means that the graticules in the graph are hidden

1 or ON means that the graticules in the graph are shown

Examples

`DISPLAY:WLAN:AD:EVM:WINDOW:TRACE:MARKER:STATE ON` shows the readout for the selected marker in the graph.

DISPlay:WLAN:AD:EVM:X:SCALe

Sets or queries the horizontal scale (time) axis range in the display.

Conditions

Measurement views: 802.11ad EVM vs Time

Group

Display commands

Syntax

```
DISPlay:WLAN:AD:EVM:X:SCALe <value>
DISPlay:WLAN:AD:EVM:X:SCALe?
```

Arguments

`<value>=<NRf>`

Returns

`<NRf>` = time trace

Examples

`DISPLAY:WLAN:AD:EVM:X:SCALE 30E-3` sets the horizontal scale to 30 ms.

DISPlay:WLAN:AD:EVM:X:SCALe:AUTO (No Query Form)

Rescales the horizontal axis automatically to fit the waveform to the screen in the display.

Conditions

Measurement views: 802.11ad EVM vs Time

Group

Display commands

Syntax

```
DISPlay:WLAN:AD:EVM:X:SCALe:AUTO
```

Arguments

None

Returns

None

Examples

`DISPLAY:WLAN:AD:EVM:X:SCALE:AUTO` rescales the horizontal scale automatically to fit the waveform to the screen.

DISPlay:WLAN:AD:EVM:X:SCALe:OFFSet

Sets or queries the start of the horizontal offset value (position) of the trace in the display.

Conditions

Measurement views: 802.11ad EVM vs Time

Group

Display commands

Syntax

```
DISPlay:WLAN:AD:EVM:X:SCALe:OFFSet <value>
DISPlay:WLAN:AD:EVM:X:SCALe:OFFSet?
```

Arguments

`<value>=<NRf>`

Returns

`<NRf>` = position of trace

Examples

`DISPLAY:WLAN:AD:EVM:X:SCALE:OFFSET 150E-3` sets the horizontal offset value of the trace to 150 ms.

DISPlay:WLAN:AD:EVM:Y:SCALe

Sets or queries the vertical scale (time) axis range of the in display.

Conditions

Measurement views: 802.11ad EVM vs Time

Group

Display commands

Syntax

```
DISPlay:WLAN:AD:EVM:Y:SCALe<value>
DISPlay:WLAN:AD:EVM:Y:SCALe?
```

Arguments

`<value>=<NRf>`

Returns

`<NRf>` = vertical range

Examples

`DISPLAY:WLAN:AD:EVM:Y:SCALE 100` sets the vertical range scale to 100 dB.

DISPlay:WLAN:AD:EVM:Y:SCALe:AUTO (No Query Form)

Rescales the vertical scale automatically to fit the waveform in the display.

Conditions

Measurement views: 802.11ad EVM vs Time

Group

Display commands

Syntax

```
DISPlay:WLAN:AD:EVM:Y:SCALe:AUTO
```

Arguments

None

Returns

None

Examples

`DISPLAY:WLAN:AD:EVM:Y:SCALE:AUTO` rescales the vertical scale to fit the waveform on the screen.

DISPlay:WLAN:AD:EVM:Y:SCALe:OFFSet

Sets or queries the start of the vertical offset scale value (position) in the display.

Conditions

Measurement views: 802.11ad EVM vs Time

Group

Display commands

Syntax

```
DISPlay:WLAN:AD:EVM:Y:SCALe:OFFSet<value>
DISPlay:WLAN:AD:EVM:Y:SCALe:OFFSet?
```

Arguments

`<value>=<NRf>`

Returns

`<NRf>` = vertical offset

Examples

`DISPLAY:WLAN:AD:EVM:Y:SCALE:OFFSET 100` sets the vertical offset to 100 dB.

DISPlay:WLAN:AD:MEASview:DELete (No Query Form)

Delete (Close) the corresponding 802.11ad measurement view from the analysis window.

Conditions

Measurement views: All 802.11ad measurement views

Group

Display commands

Syntax

`DISPlay:WLAN:AD:MEASview:DELete {CONS,SUMM,EVM,STAB}`

Arguments

`CONS` selects the 802.11ad Constellation view

`SUMM` selects the 802.11ad Summary view

`EVM` selects the 802.11ad EVM vs Time view

`STAB` selects the 802.11ad Symbol Table view

Returns

See Arguments

Examples

`DISPLAY:WLAN:AD:MEASVIEW:DELETE CONS` will close the Constellation display.

DISPlay:WLAN:AD:MEASview:NEW (No Query Form)

Displays a new 802.11ad measurement view in the analysis window.

Conditions

Measurement views: All 802.11ad measurement views

Group

Display commands

Syntax

`DISPlay:WLAN:AD:MEASview:NEW {CONS,SUMM,EVM,STAB}`

Arguments

`CONS` selects the 802.11ad Constellation view

`SUMM` selects the 802.11ad Summary view

`EVM` selects the 802.11ad EVM vs Time view

`STAB` selects the 802.11ad Symbol Table view

Returns

See Arguments

Examples

`DISPLAY:WLAN:AD:MEASVIEW:NEW CONS` will display the constellation display.

DISPlay:WLAN:AD:MEASview:SELect

Selects the corresponding 802.11 ad measurement view on the analysis window. The query command returns the currently selected view.

Conditions

Measurement views: All 802.11ad measurement views

Group

Display commands

Syntax

```
DISPlay:WLAN:AD:MEASview:SELect {CONS,SUMM,EVM,STAB}
DISPlay:WLAN:AD:MEASview:SELect?
```

Arguments

`CONS` selects the 802.11ad Constellation view

`SUMM` selects the 802.11ad Summary view

`EVM` selects the 802.11ad EVM vs Time view

`STAB` selects the 802.11ad Symbol Table view

Returns

See Arguments

Examples

`DISPLAY:WLAN:AD:MEASVIEW:SELECT SUMM` selects the 802.11ad Summary view.

DISPlay:WLAN:AY:CONSte:WINDow:TRACe:GRATicule:GRID:STATe

Set or query whether to show or hide the graticule in the Constellation display.

Conditions

Measurement view: 802.11ay Analysis Constellation

Group

Display commands

Syntax

```
DISPlay:WLAN:AY:CONSte:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:WLAN:AY:CONSte:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or `0` hides the graticule on the graph.

`ON` or `1` shows the graticule on the graph.

Returns

0 indicates the graticule is hidden on the graph.

1 indicates the graticule is shown on the graph.

Examples

`DISP:WLAN:AY:CONS:WIND:TRAC:GRAT:GRID:STAT 1` enables the graticule to show.

`DISP:WLAN:AY:CONS:WIND:TRAC:GRAT:GRID:STAT?` may return `0`, which means the graticule is disabled.

DISPlay:WLAN:AY:CONSte:WINDow:TRACe:MARKer:STATe

Set or query whether to show or hide the marker readout on the graph in the Constellation display.

Conditions

Measurement view: 802.11ay Analysis Constellation

Group

Display commands

Syntax

```
DISPlay:WLAN:AY:CONSte:WINDow:TRACe:MARKer:STATe { OFF | ON | 0 | 1 }
DISPlay:WLAN:AY:CONSte:WINDow:TRACe:MARKer:STATe?
```

Arguments

`OFF` or `0` hides the marker readout on the graph.

`ON` or `1` shows the marker readout on the graph.

Returns

`OFF` or `0` means the marker readout is hidden on the graph.

`ON` or `1` means the marker readout shows on the graph.

Examples

`DISP:WLAN:AY:CONS:WIND:TRAC:MARK:STAT 1` enables marker readout to show.

`DISP:WLAN:AY:CONS:WIND:TRAC:MARK:STAT?` may return `0`, which means marker readout is disabled.

DISPlay:WLAN:AY:CONSte:ZOOM:SCALe

Set or query the scale value, which is located under the scale tab.

Conditions

Measurement view: 802.11ay Analysis Constellation

Group

Display commands

Syntax

```
DISPlay:WLAN:AY:CONSte:ZOOM:SCALe <value>
DISPlay:WLAN:AY:CONSte:ZOOM:SCALe?
```

Arguments

`<value>::=<NRf>` specifies the scale value.

Range: 0.1 to 10

Returns

`<NRf>`

Examples

`DISP:WLAN:AY:CONS:ZOOM:SCAL 5` sets the scale value to 5.

`DISP:WLAN:AY:CONS:ZOOM:SCAL?` may return `6`, indicating that 6 is the scale value.

DISPlay:WLAN:AY:CONSte:ZOOM:[SCALe]:X:OFFSet

Set or query the horizontal position value of the scale in the Constellation display.

Conditions

Measurement view: 802.11ay Analysis Constellation

Group

Display commands

Syntax

```
DISPlay:WLAN:AY:CONSte:ZOOM:[SCALe]:X:OFFSet <value>
DISPlay:WLAN:AY:CONSte:ZOOM:[SCALe]:X:OFFSet?
```

Arguments

`<value>::=<NRf>` specifies the horizontal position value.

Range: -3.5 to 3.5

Returns

`<NRf>`

Examples

`DISP:WLAN:AY:CONS:ZOOM:SCAL:X:OFFS 3.5` sets the horizontal position value of the scale to 3.5.

`DISP:WLAN:AY:CONS:ZOOM:SCAL:X:OFFS?` may return `3.5`, which is the horizontal position value of the scale.

DISPlay:WLAN:AY:CONSte:ZOOM:[SCALe]:Y:OFFSet

Set or query the vertical position of the scale.

Conditions

Measurement view: 802.11ay Analysis Constellation

Group

Display commands

Syntax

```
DISPlay:WLAN:AY:CONSte:ZOOM:[SCALe]:Y:OFFSet <value>
DISPlay:WLAN:AY:CONSte:ZOOM:[SCALe]:Y:OFFSet?
```

Arguments

`<value>::=<NRf>` specifies the vertical position value.

Range: -3.5 to 3.5

Returns

`<NRf>`

Examples

`DISP:WLAN:AY:CONS:ZOOM:Y:OFFS -3.5` sets the vertical position value of the scale to -3.5.

`DISP:WLAN:AY:CONS:ZOOM:Y:OFFS?` may return `-3.5`, which is the vertical position value of the scale.

DISPlay:WLAN:AY:DATA:RADix

Set or query the type of symbol radix present in the WLAN 802.11ay Symbol Table control panel settings.

Conditions

Measurement views: WLAN 802.11ay Symbol Table display

Group

Display commands

Syntax

```
DISPlay:WLAN:AY:DATA:RADix { DEC | BIN }
DISPlay:WLAN:AY:DATA:RADix?
```

Arguments

`DEC` sets the data radix type to decimal.

`BIN` sets the data radix type to binary.

Returns

`DEC` means that the data radix type is decimal.

`BIN` means that the data radix type is binary.

Examples

`DISP:WLAN:AY:DATA:RAD DEC` sets the data radix to decimal.

`DISP:WLAN:AY:DATA:RAD?` may return `BIN`, indicating the data radix selected is binary.

DISPlay:WLAN:AY:EVM:SCALe:AUTO (No Query Form)

Rescales both horizontal and vertical axis automatically to fit the waveform to the screen in the 802.11ay EVM display.

Conditions

Measurement views: 802.11ay Analysis EVM

Group

Display commands

Syntax

```
DISPlay:WLAN:AY:EVM:SCALe:AUTO
```

Arguments

None

Examples

`DISP:WLAN:AY:EVM:SCAL:AUTO` rescales both horizontal and vertical axis to fit the waveform to the graph.

DISPlay:WLAN:AY:EVM:WINDow:TRACe:GRATicule:GRID:STATe

Set or query whether to show or hide the graticule in the 802.11ay EVM display.

Conditions

Measurement views: 802.11ay Analysis EVM

Group

Display commands

Syntax

```
DISPlay:WLAN:AY:EVM:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 1 | 0 }
DISPlay:WLAN:AY:EVM:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or `0` hides the graticule on the graph.

`ON` or `1` shows the graticule on the graph.

Returns

`OFF` or `0` means the graticule is hidden on the graph.

`ON` or `1` means that the graticule shows on the graph.

Examples

`DISP:WLAN:AY:EVM:WIND:TRAC:GRAT:GRID:STAT OFF` hides the graticule on the graph.

`DISP:WLAN:AY:EVM:WIND:TRAC:GRAT:GRID:STAT?` may return `ON`, which means the graticule shows on the graph.

DISPlay:WLAN:AY:EVM:WINDow:TRACe:MARKer:STATe

Set or query whether to show or hide the readout for the selected marker in the 802.11ay EVM display.

Conditions

Measurement views: 802.11ay Analysis EVM

Group

Display commands

Syntax

```
DISPlay:WLAN:AY:EVM:WINDow:TRACe:MARKer:STATe { OFF | ON | 0 | 1 }
DISPlay:WLAN:AY:EVM:WINDow:TRACe:MARKer:STATe?
```

Arguments

`OFF` or `0` hides the readout for the selected marker on the graph.

`ON` or `1` shows the readout for the selected marker on the graph.

Returns

`OFF` or `0` means the readout for the selected marker is hidden on the graph.

`ON` or `1` means the readout for the selected marker shows on the graph.

Examples

`DISP:WLAN:AY:EVM:WIND:TRAC:MARK:STAT ON` shows the readout for the selected marker on the graph.

`DISP:WLAN:AY:EVM:WIND:TRAC:MARK:STAT?` may return `OFF`, which means the readout for the selected marker is hidden on the graph.

DISPlay:WLAN:AY:EVM:X:SCALe

Set or query the horizontal (time) axis range in the 802.11ay EVM display.

Conditions

Measurement views: 802.11ay Analysis EVM

Group

Display commands

Syntax

```
DISPlay:WLAN:AY:EVM:X:SCALe <value>
DISPlay:WLAN:AY:EVM:X:SCALe?
```

Arguments

`<value>::=<NRf>` specifies the horizontal (time) axis range.

Units: ms (milliseconds)

Returns

`<NRf>`

Examples

`DISP:WLAN:AY:EVM:X:SCAL 30e-3` sets the horizontal scale to 30 ms.

`DISP:WLAN:AY:EVM:X:SCAL?` may return `30e-3`, which means that the horizontal scale is 30 ms.

DISPlay:WLAN:AY:EVM:X:SCALe:AUTO (No Query Form)

Rescale the horizontal axis automatically to fit the waveform to the screen in the EVM display.

Conditions

Measurement views: 802.11ay Analysis in 802.11ay EVM

Group

Display commands

Syntax

```
DISPlay:WLAN:AY:EVM:X:SCALe:AUTO
```

Arguments

None

Examples

`DISP:WLAN:AY:EVM:X:SCAL:AUTO` rescales the horizontal scale automatically to fit the waveform to the screen.

DISPlay:WLAN:AY:EVM:X:SCALe:OFFSet

Set or query the start of the horizontal (time) trace in the 802.11ay EVM display.

Conditions

Measurement views: 802.11ay Analysis EVM

Group

Display commands

Syntax

```
DISPlay:WLAN:AY:EVM:X:SCALe:OFFSet <value>
DISPlay:WLAN:AY:EVM:X:SCALe:OFFSet?
```

Arguments

`<value>::=<NRf>` specifies the minimum horizontal value.

Units: ms (milliseconds)

Returns

`<NRf>`

Examples

`DISP:WLAN:AY:EVM:X:SCAL:OFFS 150e-3` sets the horizontal offset value to 150 ms.

`DISP:WLAN:AY:EVM:X:SCAL:OFFS?` may return `150e-3`, which means the horizontal offset value is 150 ms.

DISPlay:WLAN:AY:EVM:Y:SCALe

Set or query the vertical range in the 802.11ay EVM display.

Conditions

Measurement views: 802.11ay Analysis EVM

Group

Display commands

Syntax

```
DISPlay:WLAN:AY:EVM:Y:SCALe <value>
DISPlay:WLAN:AY:EVM:Y:SCALe?
```

Arguments

`<value>::=<NRf>` specifies the vertical scale. Units: % or dB (based on the EVM units set)

Returns

`<NRf>`

Examples

`DISP:WLAN:AY:EVM:Y:SCAL 100` sets the vertical range to 100 dB.

`DISP:WLAN:AY:EVM:Y:SCAL?` may return `100`, which means that the vertical range is 100%.

DISPlay:WLAN:AY:EVM:Y:SCALe:AUTO (No Query Form)

Rescale the vertical scale automatically to fit the waveform in the 802.11ay EVM display.

Conditions

Measurement views: 802.11ay Analysis EVM

Group

Display commands

Syntax

```
DISPlay:WLAN:AY:EVM:Y:SCALe:AUTO
```

Arguments

None

Examples

`DISP:WLAN:AY:EVM:X:SCAL:AUTO` rescales the vertical scale automatically to fit the waveform to the screen.

DISPlay:WLAN:AY:EVM:Y:SCALe:OFFSet

Set or query the start of the vertical (EVM) trace in the 802.11ay EVM display. This command corresponds to setting or query positions in the EVM display.

Conditions

Measurement views: 802.11ay Analysis EVM

Group

Display commands

Syntax

```
DISPlay:WLAN:AY:EVM:Y:SCALe:OFFSet <value>
DISPlay:WLAN:AY:EVM:Y:SCALe:OFFSet?
```

Arguments

`<value>::=<NRf>` specifies the vertical range.

Units: % or dB (based on the EVM units set)

Returns

`<NRf>`

Examples

`DISP:WLAN:AY:EVM:Y:SCAL:OFFS 0` sets the vertical offset to 0%.

`DISP:WLAN:AY:EVM:Y:SCAL:OFFS?` may return `0`, which means the vertical offset is 0 dB.

DISPlay:WLAN:AY:MEASview:DELete (No Query Form)

Deletes the corresponding WLAN 802.11ay display from the analysis window.

Conditions

Measurement views: All WLAN 802.11ay

Group

Display commands

Syntax

```
DISPlay:WLAN:AY:MEASview:DELete { CONS | SUMM | EVM | STAB }
```

Arguments

Select one of the below arguments:

Arguments	Description
CONS	Selects the WLAN 802.11ay Constellation view.
SUMM	Selects the WLAN 802.11ay Summary view.
EVM	Selects the WLAN 802.11ay EVM versus Time view.
STAB	Selects the WLAN 802.11ay Symbol Table view.

Examples

`DISP:WLAN:AY:MEAS:DEL CONS` deletes the Constellation display.

DISPlay:WLAN:AY:MEASview:NEW (No Query Form)

Shows a new WLAN 802.11ay display on the analysis window.

Conditions

Measurement views: All WLAN 802.11ay

Group

Display commands

Syntax

```
DISPlay:WLAN:AY:MEASview:NEW { CONS | SUMM | EVM | STAB }
```

Arguments

Select one of the below arguments:

Argument	Description
CONS	Selects the WLAN 802.11ay Constellation view.
SUMM	Selects the WLAN 802.11ay Summary view.
EVM	Selects the WLAN 802.11ay EVM versus Time view.
STAB	Selects the WLAN 802.11ay Symbol Table view.

Examples

`DISP:WLAN:AY:MEAS:NEW CONS` opens the Constellation display.

DISPlay:WLAN:AY:MEASview:SELect

Set or query the corresponding 802.11 ay display on the analysis window. The query command returns the currently selected view.

Conditions

Measurement views: All WLAN 802.11ay

Group

Display commands

Syntax

```
DISPlay:WLAN:AY:MEASview:SELect { CONS | SUMM | EVM | STAB }
DISPlay:WLAN:AY:MEASview:SELect?
```

Arguments

Select one of the below arguments:

Arguments

Select one of the below arguments:

Arguments	Description
CONS	Selects the WLAN 802.11ay Constellation view.
SUMM	Selects the WLAN 802.11ay Summary view.
EVM	Selects the WLAN 802.11ay EVM versus Time view.
STAB	Selects the WLAN 802.11ay Symbol Table view.

Returns

See Arguments.

Examples

`DISP:WLAN:AY:MEAS:SEL SUMM` selects the Summary view.

`DISP:WLAN:AY:MEAS:SEL?` may return `STAB`, which indicates the Symbol Table view.

DISPlay:WLAN:CONSte[:SCALe]

Sets the magnification value (zoom factor) for the WLAN constellation plot.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:CONSte[:SCALe] <NRf>
DISPlay:WLAN:CONSte[:SCALe]?
```

Arguments

Floating point number that represents the scale factor. Valid inputs are any floating point number between 0.1 and 10. A positive value zooms in and a negative value zooms out.

Returns

A floating point number indicating the current magnification value.

Examples

`DISP:WLAN:CONS:SCAL` 5.0 zooms the display out by a factor of 5.

DISPlay:WLAN:CONSte:X:OFFSet

Sets or queries the value of the offset of the horizontal axis from the center of the WLAN Constellation display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:CONSte:X:OFFSet <NRf>
DISPlay:WLAN:CONSte:X:OFFSet?
```

Arguments

Floating point number between limits which are set as a function of the current scale value guaranteed to keep the Constellation plot at least partially visible on the display. Positive inputs push the plot down (as viewed by the user) and negative inputs push the plot up. 0 recenters the plot.

Returns

The value of the offset of the X axis from the center for the Constellation plot as a floating point number.

Examples

`DISP:WLAN:CONST:X:OFFS 1` shifts the constellation display to the left by 1.

DISPlay:WLAN:CONSte:Y:OFFSet

Sets or queries the value of the offset of the vertical axis from the center for the WLAN Constellation display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:CONSte:Y:OFFSet <NRf>
DISPlay:WLAN:CONSte:Y:OFFSet?
```

Arguments

Floating point number between limits which are set as a function of the current scale value guaranteed to keep the constellation plot at least partially visible on the display. Positive inputs push the plot down (as viewed by the user) and negative inputs push the plot up. 0 re-centers the plot.

Returns

The value of the offset of the Y axis from the center for the Constellation display as a floating point number.

Examples

`DISP:WLAN:CONST:Y:OFFS 1` shifts the constellation display up by 1.

DISPlay:WLAN:CRESponse:AUTO (No Query Form)

Automatically rescales the horizontal and vertical values for the best display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

`DISPlay:WLAN:CRESponse:AUTO`

Arguments

None

Examples

`DISP:WLAN:CRES:AUTO` automatically rescales the horizontal and vertical values.

DISPlay:WLAN:CRESponse:FREQuency:AUTO (No Query Form)

Automatically rescales the horizontal subcarrier (frequency) start and stop values to fit the waveform to the screen in the WLAN Channel Response display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

`DISPlay:WLAN:CRESponse:FREQuency:AUTO`

Arguments

None

Examples

`DISP:WLAN:CRES:FREQ:AUTO` automatically rescales the horizontal subcarrier (frequency) start and stop values to fit the waveform to the screen.

DISPlay:WLAN:CRESponse:FREQuency:STARt

Specifies or queries the horizontal start value (left edge) on the WLAN Channel Response display. The units are either Frequency (Hz) or Subcarriers, and are set using the command `[SENSe]:WLAN:UNIT:FREQuency`.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:CRESponse:FREQuency:STARt <NRf>
DISPlay:WLAN:CRESponse:FREQuency:STARt?
```

Arguments

Floating point number that specifies the horizontal start value (left edge) on the graph.

Returns

The returned value indicates the Subcarrier (for non-b standards only) or frequency shown at the left edge of the display.

Examples

`DISP:WLAN:CRES:FREQ:START 2.41e9` sets the horizontal start value (left edge) to 2.41 GHz (assuming the units have been set to Frequency).

DISPlay:WLAN:CRESponse:FREQuency:STOP

Specifies or queries the horizontal stop value (right edge) on the WLAN Channel Response display. The units are either Frequency (Hz) or Subcarriers, and are set using the command `[SENSe]:WLAN:UNIT:FREQuency`.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:CRESponse:FREQuency:STOP <NRf>
DISPlay:WLAN:CRESponse:FREQuency:STOP?
```

Arguments

Floating point number that specifies the horizontal stop value (right edge) on the display.

Examples

`DISP:WLAN:CRES:FREQ:STOP 8.1250e9` sets the horizontal stop value to 8.1250 GHz (assuming the units have been set to Frequency).

DISPlay:WLAN:CRESponse:MAGNitude:AUTO (No Query Form)

Rescales the vertical magnitude offset (position) and scale values to automatically to fit the waveform to the screen in the WLAN Channel Response magnitude graph.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:CRESponse:MAGNitude:AUTO
```

Arguments

None

Examples

`DISP:WLAN:CRES:MAGN:AUTO` rescales the vertical magnitude offset (position) and scale values automatically to fit the waveform to the screen.

DISPlay:WLAN:CRESponse:MAGNitude:OFFSet

Sets or queries the vertical offset (position in the center of the vertical axis) in the WLAN Channel Response magnitude graph.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:CRESponse:MAGNitude:OFFSet <NRf>
DISPlay:WLAN:CRESponse:MAGNitude:OFFSet?
```

Arguments

Floating point number that specifies the vertical offset in dB.

Examples

`DISP:WLAN:CRES:MAGN:OFFS 5` sets the vertical offset to 5 dB.

DISPlay:WLAN:CRESponse:MAGNitude[:SCALe]

Sets or queries the value of the vertical scale for the WLAN Channel Response magnitude display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:CRESponse:MAGNitude[:SCALe] <NRf>
DISPlay:WLAN:CRESponse:MAGNitude[:SCALe]?
```

Arguments

Floating point number that represents the vertical scale value in dB.

Examples

`DISP:WLAN:CRES:MAGN:SCAL 5.0` sets the vertical scale to 5 dB.

DISPlay:WLAN:CRESponse:MARKer:SHOW:STATe

Shows or hides the marker readout on the WLAN Channel Response display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:CRESponse:MARKer:SHOW:STATe {0|1|OFF|ON}
DISPlay:WLAN:CRESponse:MARKer:SHOW:STATe?
```

Arguments

`ON` or `1` specifies to show the marker readout.

`OFF` or `0` specifies to turn the marker readout off.

Examples

`DISP:WLAN:CRES:MARK:SHOW:STATE ON` specifies to turn the marker readout on.

DISPlay:WLAN:CRESponse:PHASe:AUTO (No Query Form)

Rescales the vertical phase value automatically to fit the waveform to the screen in the WLAN Channel Response display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:CRESponse:PHASe:AUTO
```

Arguments

None

Examples

`DISP:WLAN:CRES:PHAS:AUTO` rescales the vertical phase value automatically to fit the waveform to the screen.

DISPlay:WLAN:CRESponse:PHASe:OFFSet

Sets or queries the vertical offset (position in the center of the vertical axis) in the WLAN Channel Response phase graph.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:CRESponse:PHASe:OFFSet <NRf>
DISPlay:WLAN:CRESponse:PHASe:OFFSet?
```

Arguments

Floating point number that specifies the vertical offset (position) in degrees.

Examples

`DISPLAY:WLAN:CRESPONSE:PHASE:OFFSET -14.5` sets the vertical offset value to -14.5º in the WLAN Channel Response phase graph.

DISPlay:WLAN:CRESponse:PHASe[:SCALe]

Sets or queries the phase value for the WLAN Channel Response display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:CRESponse:PHASe[:SCALe] <NRf>
DISPlay:WLAN:CRESponse:PHASe[:SCALe]?
```

Arguments

Floating point number that represents the phase value in degrees.

Examples

`DISP:WLAN:CRES:PHASE:SCAL 5` sets the vertical scale to 5 degrees for the display.

DISPlay:WLAN:CRESponse:WINDow:SELect:PLOT

Specifies or queries which graphs are displayed in the WLAN Channel Response display view. Select from Magnitude, Phase, or Both.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:CRESponse:WINDow:SELect:PLOT {MAGNitude|PHASe|BOTH}
DISPlay:WLAN:CRESponse:WINDow:SELect:PLOT?
```

Arguments

`MAGNitude` displays the Magnitude graph.

`PHASe` displays the Phase graph.

`BOTH` displays both graphs.

Examples

`DISP:WLAN:CRES:WIND:SEL:PLOT PHAS` displays the Phase graph on the WLAN Channel Response analysis display.

DISPlay:WLAN:CRESponse:WINDow:TRACe:GRATicule:GRID:STATe

Sets or queries whether to show the graticule grid on the WLAN Channel Response analysis display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:CRESponse:WINDow:TRACe:GRATicule:GRID:STATe {0|1|OFF|ON}
DISPlay:WLAN:CRESponse:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`ON` or `1` specifies to show the graticule.

`OFF` or `0` specifies to turn it off.

Examples

`SENSE:WLAN:CRES:WIND:TRAC:GRAT:GRID:STATE ON` specifies to turn the graticule on.

DISPlay:WLAN:EVM:FREQuency:AUTO (No Query Form)

Automatically rescales the horizontal subcarrier (frequency) start and stop values to fit the waveform to the screen in the WLAN EVM display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

`DISPlay:WLAN:EVM:FREQuency:AUTO`

Arguments

None

Examples

`DISP:WLAN:EVM:FREQ:AUTO` automatically rescales the horizontal Subcarrier (frequency) values to fit the waveform to the screen.

DISPlay:WLAN:EVM:FREQuency:RESet (No Query Form)

Automatically resets the horizontal subcarrier (frequency) start and stop values for the WLAN EVM display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

`DISPlay:WLAN:EVM:FREQuency:RESet`

Arguments

None

Examples

`DISP:WLAN:EVM:FREQ:RES` automatically resets the horizontal subcarrier start and stop values for the WLAN EVM display.

DISPlay:WLAN:EVM:FREQuency:STARt

Specifies or queries the horizontal start value (left edge) on the WLAN EVM display. The units are either Frequency (Hz) or Subcarriers, and are set using the command `[SENSe]:WLAN:UNIT:FREQuency`.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:EVM:FREQuency:STARt <NRf>
DISPlay:WLAN:EVM:FREQuency:STARt?
```

Arguments

Floating point number that specifies the horizontal start value (left edge) on the graph.

Returns

The returned value indicates the Subcarrier (for non-b standards only) or frequency shown at the left edge of the display.

Examples

`DISP:WLAN:EVM:FREQ:START 2.41e9` sets the horizontal start value (left edge) to 2.41e9 MHz (assuming the units have been set to Frequency).

DISPlay:WLAN:EVM:FREQuency:STOP

Specifies or queries the horizontal stop value (right edge) on the WLAN EVM display. The units are either Frequency (Hz) or Subcarriers, and are set using the command `[SENSe]:WLAN:UNIT:FREQuency`.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:EVM:FREQuency:STOP <NRf>
DISPlay:WLAN:EVM:FREQuency:STOP?
```

Arguments

Floating point number that specifies the horizontal stop value (right edge) on the display.

Examples

`DISP:WLAN:EVM:FREQ:STOP 8.1250` sets the horizontal stop value to 8.1250 MHz (assuming the units have been set to Frequency).

DISPlay:WLAN:EVM:MARKer:SHOW:STATe

Shows or hides the Marker readout on the WLAN EVM display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:EVM:MARKer:SHOW:STATe {0|1|OFF|ON}
DISPlay:WLAN:EVM:MARKer:SHOW:STATe?
```

Arguments

`ON` or 1 specifies to show the Marker readout.

`OFF` or 0 specifies to turn the Marker readout off.

Examples

`DISP:WLAN:EVM:MARK:SHOW:STATE ON` specifies to turn the Marker readout on.

DISPlay:WLAN:EVM[:SCALe]:AUTO (No Query Form)

Automatically adjusts all vertical and horizontal values to provide the best display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:EVM[:SCALe]:AUTO
```

Arguments

None

Examples

`DISP:WLAN:EVM:SCALE:AUTO` automatically adjusts all vertical and horizontal values to provide the best display.

DISPlay:WLAN:EVM:TIME:AUTO (No Query Form)

Sets the horizontal start and stop values to fit the waveform to the WLAN EVM display, and also enables Auto horizontal scaling so the waveform will remain properly scaled as burst lengths change.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:EVM:TIME:AUTO
```

Arguments

None

Examples

DISP:WLAN:EVM:TIME:AUTO sets the horizontal start and stop values to fit the waveform to the display, and also enables Auto horizontal scaling.

DISPlay:WLAN:EVM:TIME:AUTO:STATe

When enabled, sets the horizontal symbol scale value automatically for the WLAN EVM display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:EVM:TIME:AUTO:STATe {0|1|OFF|ON}
DISPlay:WLAN:EVM:TIME:AUTO:STATe?
```

Arguments

ON or 1 sets the horizontal symbol scale value automatically.

OFF or 0 specifies to stop setting the horizontal symbol scale value automatically.

Examples

DISP:WLAN:EVM:TIME:AUTO:STATE ON specifies to the horizontal symbol scale value automatically.

DISPlay:WLAN:EVM:TIME:RESet (No Query Form)

Resets the horizontal symbol start and stop values for the WLAN EVM display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:EVM:TIME:RESet
```

Arguments

None

Examples

DISP:WLAN:EVM:TIME:RES resets the horizontal symbol start and stop values for the WLAN EVM display.

DISPlay:WLAN:EVM:TIME:STARt

Sets or queries the horizontal symbol start value for the WLAN EVM display. The units are either Symbols or Seconds, and are set using the command `[SENSe]:WLAN:UNIT:TIME`.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:EVM:TIME:STARt <NRf>
DISPlay:WLAN:EVM:TIME:STARt?
```

Arguments

Floating point number.

Examples

`DISP:WLAN:EVM:TIME:START 40` sets the horizontal symbol start value of the WLAN EVM display to 40 Symbols (assuming the units have been set to Symbols).

DISPlay:WLAN:EVM:TIME:STOP

Sets or queries the horizontal symbol stop value for the WLAN EVM display. The units are either Symbols or Seconds, and are set using the command `[SENSe]:WLAN:UNIT:TIME`.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:EVM:TIME:STOP <NRf>
DISPlay:WLAN:EVM:TIME:STOP?
```

Arguments

Floating point number.

Examples

`DISP:WLAN:EVM:TIME:STOP 804.96e-6` sets the horizontal symbol stop value for the WLAN EVM display to 804.960 µs (assuming the units have been set to Seconds).

DISPlay:WLAN:EVM:WINDow:SELect:PLOT

Specifies or queries which graphs are displayed in the WLAN EVM display. Select from Symbols, Subcarriers, or Both.

To set the Time units, use `[SENSe]:WLAN:UNIT:TIME`. To set the Frequency units, use `[SENSe]:WLAN:UNIT:FREQuency`. To set the Radix, use `[SENSe]:WLAN:RADix`.

NOTE: Use this command only with multicarrier standards and not singlecarrier (802.11b).

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:EVM:WINDow:SELect:PLOT {SYMBols|SCARrier|BOTH}
DISPlay:WLAN:EVM:WINDow:SELect:PLOT?
```

Arguments

`SYMBols` displays the Symbols graph.

`SCARrier` displays the Subcarriers graph.

`BOTH` displays both graphs.

Examples

`DISP:WLAN:EVM:WIND:SEL:PLOT SCAR` displays the Subcarrier graph on the WLAN EVM display.

DISPlay:WLAN:EVM:WINDow:TRACe:GRATicule:GRID:STATe

Shows or hides the graticule grid on the screen.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:EVM:WINDow:TRACe:GRATicule:GRID:STATe {0|1|OFF|ON}
DISPlay:WLAN:EVM:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`ON` or 1 specifies to show the graticule.

`OFF` or 0 specifies to turn it off.

Examples

`DISP:WLAN:EVM:WIND:TRAC:GRAT:GRID:STATE ON` specifies to turn the graticule on.

DISPlay:WLAN:EVM:Y:AUTO (No Query Form)

Automatically sets the vertical scale and position values for the WLAN EVM display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:EVM:Y:AUTO
```

Arguments

None

Examples

`DISP:WLAN:EVM:Y:AUTO` automatically sets the vertical scale and position values for the WLAN EVM display.

DISPlay:WLAN:EVM:Y:OFFSet

Sets or queries the value of the offset from the center (vertical position), in percent terms, for the WLAN EVM display. To set the vertical scale, use the command `DISPlay:WLAN:EVM:Y[:SCALe]`.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:EVM:Y:OFFSet <NRf>
DISPlay:WLAN:EVM:Y:OFFSet?
```

Arguments

Floating point number between limits which are set as a function of the current scale value guaranteed to keep the EVM plot at least partially visible on the display. Positive inputs push the plot down (as viewed by the user) and negative inputs push the plot up. 0 re-centers the plot.

Returns

The value of the offset from the center of the Y axis for the Error Magnitude Vector plot as a floating point number.

Examples

`DISP:WLAN:EVM:Y:OFFS 150` sets the offset from the center (vertical position) of the EVM plot to 150%.

DISPlay:WLAN:EVM:Y:RESet (No Query Form)

Resets the vertical scale and position (offset) values for the WLAN EVM display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:EVM:Y:RESet
```

Arguments

None

Examples

`DISP:WLAN:EVM:Y:RES` specifies to reset the vertical scale and position (offset) values for the WLAN EVM display.

DISPlay:WLAN:EVM:Y[:SCALe]

Sets or queries the vertical scale value for the WLAN EVM display. To set the vertical position (offset) value, use the command DISPlay:WLAN:EVM:Y:OFFSet (on page 1027).

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:EVM:Y[:SCALe] <NRf>
DISPlay:WLAN:EVM:Y[:SCALe]?
```

Arguments

Floating point number that specifies the vertical scale value for the WLAN EVM display.

Examples

`DISP:WLAN:EVM:Y:SCALE 10` sets the vertical scale value for the WLAN EVM display to 10.

DISPlay:WLAN:FLATness:AUTO (No Query Form)

Automatically rescales the horizontal and vertical axes for the best display in the WLAN Spectral Flatness view.

NOTE: WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:FLATness:AUTO
```

Arguments

None

Examples

`DISP:WLAN:FLAT:AUTO` automatically rescales the horizontal and vertical axes for the best display.

DISPlay:WLAN:FLATness:X:AUTO (No Query Form)

Rescales the horizontal axis automatically in the WLAN Spectral Flatness display.

NOTE: WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:FLATness:X:AUTO
```

Arguments

None

Examples

`DISP:WLAN:FLAT:X:AUTO` rescales the horizontal axis automatically for the best display.

DISPlay:WLAN:FLATness:X:OFFSet

Sets or queries the value of the offset of the horizontal axis from the center of the WLAN Spectral Flatness display.

NOTE: WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:FLATness:X:OFFSet <NRf>
DISPlay:WLAN:FLATness:X:OFFSet?
```

Arguments

Floating point number between limits which are set as a function of the current scale value guaranteed to keep the plot at least partially visible on the display. Positive inputs push the plot down (as viewed by the user) and negative inputs push the plot up. 0 recenters the plot.

Returns

The value of the offset of the X axis from the center of the plot as a floating point number.

Examples

`DISP:WLAN:FLAT:X:OFFS 150` shifts the display to the left by 150%.

DISPlay:WLAN:FLATness:X[:SCALe]

Sets or queries the horizontal scale value in the WLAN Spectral Flatness display. The units are Frequency or Subcarrier. To set the units, use `[SENSe]:WLAN:UNIT:FREQuency`.

NOTE: WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:FLATness:X[:SCALe] <NRf>
DISPlay:WLAN:FLATness:X[:SCALe]?
```

Arguments

Floating point number that represents the horizontal scale.

Examples

`DISP:WLAN:FLAT:X:SCALE 32` sets the horizontal scale value to 32.

DISPlay:WLAN:FLATness:Y:AUTO (No Query Form)

Automatically rescales the vertical axis scale and position values in the WLAN Spectral Flatness display.

NOTE: WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:FLATness:Y:AUTO
```

Arguments

None

Examples

`DISP:WLAN:FLAT:Y:AUTO` automatically selects the vertical scale and position values.

DISPlay:WLAN:FLATness:Y:OFFSet

Sets or queries the value of the offset from the center (vertical position), in percent terms, for the WLAN Spectral Flatness display. To set the Vertical Scale, use the command DISPlay:WLAN:FLATness:Y[:SCALe] (on page 1031).

NOTE: WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:FLATness:Y:OFFSet <NRf>
DISPlay:WLAN:FLATness:Y:OFFSet?
```

Arguments

Floating point number between limits which are set as a function of the current scale value guaranteed to keep the plot at least partially visible on the display. Positive inputs push the plot down (as viewed by the user) and negative inputs push the plot up. 0 re-centers the plot.

Returns

The value of the offset from the center of the Y axis for the plot as a floating point number.

Examples

`DISP:WLAN:FLAT:Y:OFFS 150` sets the offset from the center (vertical position) of the plot to 150 %.

DISPlay:WLAN:FLATness:Y[:SCALe]

Sets or queries the vertical scale value for the WLAN Spectral Flatness display. To set the vertical position (offset) value, use the command `DISPlay:WLAN:FLATness:Y:OFFSet`.

NOTE: WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:FLATness:Y[:SCALe] <NRf>
DISPlay:WLAN:FLATness:Y[:SCALe]?
```

Arguments

Floating point number that specifies the vertical scale value.

Examples

`DISP:WLAN:FLAT:Y:SCALE 10` sets the vertical scale value to 10.

DISPlay:WLAN:MEASview:DELete (No Query Form)

Deletes the selected measurement view in the WLAN display.

If you attempt to delete a view that is currently displayed on screen, the error (-200, "Execution error; Measurement is already running") will be returned.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:MEASview:DELete {SUMMary|FLATness|CONStellation|EVM|MERRor|PERRor|CRESponse|STABle|
    PVTime}
```

Arguments

The following table lists the arguments.

Argument	WLAN measurement view
SUMMary	Summary measurements
FLATness	Spectral Flatness measurement
CONStellation	Constellation measurement

Examples

`DISP:WLAN:MEAS:DEL PERR` deletes the Phase Error measurement view.

DISPlay:WLAN:MEASview:NEW (No Query Form)

Displays a new measurement view in the WLAN display.

If you attempt to open a view that is currently displayed on screen, the error (-200, "Execution error; Measurement is already running") will be returned.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:MEASview:NEW {SUMMary|FLATness|CONStellation|EVM|MERRor|PERRor|CRESponse|STABle|
    PVTime}
```

Arguments

The following table lists the arguments.

Argument	WLAN measurement view
SUMMary	Summary measurements
FLATness	Spectral Flatness measurement
CONStellation	Constellation measurement

Examples

`DISP:WLAN:MEASVIEW:NEW CONSTELLATION` creates the Constellation measurement view.

DISPlay:WLAN:MEASview:SELect

Selects a measurement view on the screen in the WLAN measurements. The query command returns the currently selected view.

If you attempt to select a view that is not displayed on screen, the error (-200, "Execution error; Measurement is already running") will be returned.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:MEASview:SELect {SUMMary|FLATness|CONStellation|EVM|MERRor|PERRor|CRESponse|STABle|
    PVTime}
DISPlay:WLAN:MEASview:SELect?
```

Arguments

The following table lists the arguments.

The following table lists the arguments.

Argument	WLAN measurement view
SUMMary	Summary measurements
FLATness	Spectral Flatness measurement
CONStellation	Constellation measurement
EVM	Error Vector Magnitude measurement

Argument	WLAN measurement view
MERRor	Magnitude Error vs. Time
PERRor	Phase Error vs. Time
CRESponse	Channel Response measurement
STABle	Symbol Table
PVTime	Power Vs. Time

Examples

`DISP:WLAN:MEASVIEW:SEL CONSTELLATION` selects the Constellation measurement view.

`DISP:WLAN:MEASVIEW:SEL?` might return FLAT, indicating that the Spectral Flatness measurement view is the currently selected view.

DISPlay:WLAN:MERRor:FREQuency:AUTO (No Query Form)

Automatically rescales the horizontal subcarrier (frequency) start and stop values to fit the waveform to the screen in the WLAN Magnitude Error display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

`DISPlay:WLAN:MERRor:FREQuency:AUTO`

Arguments

None

Examples

`DISPlay:WLAN:MERRor:FREQuency:AUTO` automatically rescales the horizontal Subcarrier (frequency) values to fit the waveform to the screen.

DISPlay:WLAN:MERRor:FREQuency:RESet (No Query Form)

Automatically resets the horizontal subcarrier (frequency) start and stop values for the WLAN Magnitude Error display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:MERRor:FREQuency:RESet
```

Arguments

None

Examples

`DISP:WLAN:MERR:FREQ:RES` automatically resets the horizontal subcarrier start and stop values for the WLAN Magnitude Error display.

DISPlay:WLAN:MERRor:FREQuency:STARt

Specifies or queries the horizontal start value (left edge) on the WLAN Magnitude Error display. The units are either Frequency (Hz) or Subcarriers, and are set using the command [SENSe]:WLAN:UNIT:FREQuency (on page 2480).

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:MERRor:FREQuency:STARt <NRf>
DISPlay:WLAN:MERRor:FREQuency:STARt?
```

Arguments

Floating point number that specifies the horizontal start value (left edge) on the graph.

Returns

The returned value indicates the Subcarrier (for non-b standards only) or frequency shown at the left edge of the display.

Examples

`DISP:WLAN:MERR:FREQ:START 2.41e9` sets the horizontal start value (left edge) to 2.41e9 MHz (assuming the units have been set to Frequency).

DISPlay:WLAN:MERRor:FREQuency:STOP

Specifies or queries the horizontal stop value (right edge) on the WLAN Magnitude Error display. The units are either Frequency (Hz) or Subcarriers, and are set using the command [SENSe]:WLAN:UNIT:FREQuency (on page 2480).

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:MERRor:FREQuency:STOP <NRf>
DISPlay:WLAN:MERRor:FREQuency:STOP?
```

Arguments

Floating point number that specifies the horizontal stop value (right edge) on the display.

Examples

`DISP:WLAN:MERR:FREQ:STOP 8.1250` sets the horizontal stop value to 8.1250 MHz (assuming the units have been set to Frequency).

DISPlay:WLAN:MERRor:MARKer:SHOW:STATe

Shows or hides the marker readout on the WLAN Magnitude Error display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:MERRor:MARKer:SHOW:STATe {0|1|OFF|ON}
DISPlay:WLAN:MERRor:MARKer:SHOW:STATe?
```

Arguments

`ON` or 1 specifies to show the marker readout.

`OFF` or 0 specifies to turn the marker readout off.

Examples

`DISP:WLAN:MERR:MARK:SHOW:STATE ON` specifies to turn the marker readout on.

DISPlay:WLAN:MERRor[:SCALe]:AUTO (No Query Form)

Automatically adjusts all vertical and horizontal values to provide the best display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:MERRor[:SCALe]:AUTO
```

Arguments

None

Examples

`DISP:WLAN:MERR:SCALE:AUTO` automatically adjusts all vertical and horizontal values to provide the best display.

DISPlay:WLAN:MERRor:TIME:AUTO (No Query Form)

Sets the horizontal start and stop values to fit the waveform to the WLAN Magnitude Error display, and also enables Auto horizontal scaling so the waveform will remain properly scaled as burst lengths change.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:MERRor:TIME:AUTO
```

Arguments

None

Examples

`DISP:WLAN:MERR:TIME:AUTO` sets the horizontal start and stop values to fit the waveform to the display, and also enables Auto horizontal scaling.

DISPlay:WLAN:MERRor:TIME:AUTO:STATe

When enabled, automatically sets the horizontal symbol scale value for the WLAN MERR display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:MERRor:TIME:AUTO:STATe {0|1|OFF|ON}
DISPlay:WLAN:MERRor:TIME:AUTO:STATe?
```

Arguments

`ON` or 1 sets the horizontal symbol scale value automatically.

`OFF` or 0 specifies to stop setting the horizontal symbol scale value automatically.

Examples

`DISP:WLAN:MERR:TIME:AUTO:STATE ON` specifies to the horizontal symbol scale value automatically.

DISPlay:WLAN:MERRor:TIME:RESet (No Query Form)

Resets the horizontal symbol start and stop values for the WLAN Magnitude Error display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:MERRor:TIME:RESet
```

Arguments

None

Examples

`DISP:WLAN:MERR:TIME:RES` resets the horizontal symbol start and stop values for the WLAN Magnitude Error display.

DISPlay:WLAN:MERRor:TIME:STARt

Sets or queries the horizontal symbol start value for the WLAN Magnitude Error display. The units are either Symbols or Seconds, and are set using the command `[SENSe]:WLAN:UNIT:TIME`.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:MERRor:TIME:STARt <NRf>
DISPlay:WLAN:MERRor:TIME:STARt?
```

Arguments

Floating point number.

Examples

`DISP:WLAN:MERR:TIME:START 40` sets the horizontal symbol start value of the WLAN Magnitude Error display to 40 Symbols (assuming the units have been set to Symbols).

DISPlay:WLAN:MERRor:TIME:STOP

Sets or queries the horizontal symbol stop value for the WLAN Magnitude Error display. The units are either Symbols or Seconds, and are set using the command `[SENSe]:WLAN:UNIT:TIME`.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:MERRor:TIME:STOP <NRf>
DISPlay:WLAN:MERRor:TIME:STOP?
```

Arguments

Floating point number.

Examples

`DISP:WLAN:MERR:TIME:STOP 804.96e-6` sets the horizontal symbol stop value for the WLAN Magnitude Error display to 804.960 µs (assuming the units have been set to Seconds).

DISPlay:WLAN:MERRor:WINDow:SELect:PLOT

Specifies or queries which graphs are displayed in the WLAN Magnitude Error display. Select from Symbols, Subcarriers, or Both.

To set the Time units, use [SENSe]:WLAN:UNIT:TIME (on page 2481). To set the Frequency units, use [SENSe]:WLAN:UNIT:FREQuency (on page 2480). To set the Radix, use [SENSe]:WLAN:RADix (on page 2471).

NOTE: Use this command only with multicarrier standards and not singlecarrier (802.11b).

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:MERRor:WINDow:SELect:PLOT{SYMBols|SCARrier|BOTH}
DISPlay:WLAN:MERRor:WINDow:SELect:PLOT?
```

Arguments

`SYMBols` displays the Symbols graph.

`SCARrier` displays the Subcarriers graph.

`BOTH` displays both graphs.

Examples

`DISP:WLAN:MERR:WIND:SEL:PLOT SCAR` displays the Subcarrier graph on the WLAN Magnitude Error display.

DISPlay:WLAN:MERRor:WINDow:TRACe:GRATicule:GRID:STATe

Shows or hides the graticule grid on the display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:MERRor:WINDow:TRACe:GRATicule:GRID:STATe {0|1|OFF|ON}
DISPlay:WLAN:MERRor:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`ON` or 1 specifies to show the graticule.

`OFF` or 0 specifies to turn it off.

Examples

`DISP:WLAN:MERR:WIND:TRAC:GRAT:GRID:STATE ON` specifies to turn the graticule on.

DISPlay:WLAN:MERRor:Y:AUTO (No Query Form)

Automatically sets the vertical scale and position values for the WLAN Magnitude Error display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:MERRor:Y:AUTO
```

Arguments

None

Examples

`DISP:WLAN:MERR:Y:AUTO` automatically sets the vertical scale and position values for the WLAN Magnitude Error display.

DISPlay:WLAN:MERRor:Y:OFFSet

Sets or queries the offset from the center (vertical position), in percent terms, for the WLAN Magnitude Error display. To set the vertical scale, use the command `DISPlay:WLAN:EVM:Y[:SCALe]`.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:MERRor:Y:OFFSet <NRf>
DISPlay:WLAN:MERRor:Y:OFFSet?
```

Arguments

Floating point number between limits which are set as a function of the current scale value guaranteed to keep the Magnitude Error plot at least partially visible on the display. Positive inputs push the plot down (as viewed by the user) and negative inputs push the plot up. 0 re-centers the plot.

Returns

The offset from the center of the Y axis for the Magnitude Error plot as a floating point number.

Examples

`DISP:WLAN:MERR:Y:OFFS 150` sets the offset from the center of the vertical axis of the Magnitude Error plot to 150%.

DISPlay:WLAN:MERRor:Y:RESet (No Query Form)

Resets the vertical scale and position (offset) for the WLAN Magnitude Error display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:MERRor:Y:RESet
```

Arguments

None

Examples

`DISP:WLAN:MERR:Y:RES` specifies to reset the vertical scale and position (offset) values for the WLAN Magnitude Error display.

DISPlay:WLAN:MERRor:Y[:SCALe]

Sets or queries the vertical scale value for the WLAN Magnitude Error display. To set the vertical position (offset) value, use the command `DISPlay:WLAN:EVM:Y:OFFSet`.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:MERRor:Y[:SCALe] <NRf>
DISPlay:WLAN:MERRor:Y[:SCALe]?
```

Arguments

Floating point number that specifies the vertical scale value for the WLAN Magnitude Error display.

Examples

`DISP:WLAN:MERR:Y:SCALE 10` sets the vertical scale for the WLAN Magnitude Error display to 10.

DISPlay:WLAN:PERRor:FREQuency:AUTO (No Query Form)

Automatically rescales the horizontal subcarrier (frequency) start and stop values to fit the waveform to the screen in the WLAN Phase Error display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

`DISPlay:WLAN:PERRor:FREQuency:AUTO`

Arguments

None

Examples

`DISP:WLAN:PERR:FREQ:AUTO` automatically rescales the horizontal Subcarrier (frequency) start and stop values to fit the waveform to the screen.

DISPlay:WLAN:PERRor:FREQuency:RESet (No Query Form)

Automatically resets the horizontal subcarrier (frequency) start and stop values for the WLAN Phase Error display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

`DISPlay:WLAN:PERRor:FREQuency:RESet`

Arguments

None

Examples

`DISP:WLAN:PERR:FREQ:RES` automatically resets the horizontal Subcarrier start and stop values for the WLAN Phase Error display.

DISPlay:WLAN:PERRor:FREQuency:STARt

Specifies or queries the horizontal start value (left edge) on the WLAN Phase Error display. The units are either Frequency (Hz) or Subcarriers, and are set using the command `[SENSe]:WLAN:UNIT:FREQuency`.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PERRor:FREQuency:STARt <NRf>
DISPlay:WLAN:PERRor:FREQuency:STARt?
```

Arguments

Floating point number that specifies the horizontal start value (left edge) on the display.

Returns

The readout indicates the Subcarrier (for non-b standards only) or frequency shown at the left edge of the display.

Examples

`DISP:WLAN:PERR:FREQ:START -8.1250` sets the horizontal start value to -8.1250 MHz (assuming the units have been set to Frequency).

DISPlay:WLAN:PERRor:FREQuency:STOP

Specifies or queries the horizontal stop value (right edge) on the WLAN Phase Error display. The units are either Frequency (Hz) or Subcarriers, and are set using the command `[SENSe]:WLAN:UNIT:FREQuency`.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PERRor:FREQuency:STOP <NRf>
DISPlay:WLAN:PERRor:FREQuency:STOP?
```

Arguments

Floating point number that specifies the horizontal stop value (right edge) on the display.

Examples

`DISP:WLAN:PERR:FREQ:STOP 8.1250` sets the horizontal stop value to 8.1250 MHz (assuming the units have been set to Frequency).

DISPlay:WLAN:PERRor:MARKer:SHOW:STATe

Show or hides the Marker readout on the WLAN Phase Error display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PERRor:MARKer:SHOW:STATe {0|1|OFF|ON}
DISPlay:WLAN:PERRor:MARKer:SHOW:STATe?
```

Arguments

`ON` or 1 specifies to show the Marker readout.

`OFF` or 0 specifies to turn the Marker readout off.

Examples

`DISP:WLAN:PERR:MARK:SHOW:STATE ON` specifies to turn the Marker readout on.

DISPlay:WLAN:PERRor[:SCALe]:AUTO (No Query Form)

Automatically adjusts all vertical and horizontal settings to provide the best display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PERRor[:SCALe]:AUTO
```

Arguments

None

Examples

`DISP:WLAN:PERRor:SCALE:AUTO` automatically adjusts all vertical and horizontal settings to provide the best display.

DISPlay:WLAN:PERRor:TIME:AUTO (No Query Form)

Automatically sets the horizontal symbol start and stop values to fit the waveform to the screen for the WLAN Phase Error vs. Symbol plot. In addition, sets the state for this feature to ON.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PERRor:TIME:AUTO
```

Arguments

None

Examples

`DISP:WLAN:PERR:TIME:AUTO` sets the instrument to automatically select the horizontal symbol start and stop values to fit the waveform to the screen, and turns the AUTO feature on.

DISPlay:WLAN:PERRor:TIME:AUTO:STATe

When enabled, automatically sets the horizontal symbol scale value for the WLAN Phase Error display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PERRor:TIME:AUTO:STATe {0|1|OFF|ON}
DISPlay:WLAN:PERRor:TIME:AUTO:STATe?
```

Arguments

`ON` or 1 specifies to set the Analysis Time value automatically.

`OFF` or 0 specifies to turn stop setting the Analysis Time value automatically.

Examples

`DISP:WLAN:PERR:TIME:AUTO:STATE ON` specifies to set the Analysis Time value automatically, based on the requirements of the selected display

DISPlay:WLAN:PERRor:TIME:RESet (No Query Form)

Resets the horizontal symbol start and stop values for the WLAN Phase Error Average vs. Symbol display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PERRor:TIME:RESet
```

Arguments

None

Examples

DISP:WLAN:PERR:TIME:RES resets the horizontal symbol start and stop values for the WLAN Phase Error Average vs. Symbol display.

DISPlay:WLAN:PERRor:TIME:STARt

Sets or queries the horizontal symbol start value for the WLAN Phase Error Average vs. Symbol display. The units are either Symbols or Seconds, and are set using the command [SENSe]:WLAN:UNIT:TIME.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PERRor:TIME:STARt <NRf>
DISPlay:WLAN:PERRor:TIME:STARt?
```

Arguments

Floating point number.

Examples

DISP:WLAN:PERR:TIME:START 40 sets the horizontal symbol start value of the Average vs. Symbol display to 40 Symbols (assuming the units have been set to Symbols).

DISPlay:WLAN:PERRor:TIME:STOP

Sets or queries the horizontal symbol stop value for the WLAN Phase Error Average vs. Symbol display. The units are either Symbols or Seconds, and are set using the command [SENSe]:WLAN:UNIT:TIME.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PERRor:TIME:STOP <NRf>
DISPlay:WLAN:PERRor:TIME:STOP?
```

Arguments

Floating point number.

Examples

DISP:WLAN:PERR:TIME:STOP 804.960 sets the horizontal symbol stop value for the Average vs. Symbol display to 804.960 µs (assuming the units have been set to Seconds).

DISPlay:WLAN:PERRor:WINDow:SELect:PLOT

Specifies or queries which displays are displayed in the WLAN Phase Error analysis display (Symbols, Subcarriers, or Both).

To set the Time units, use [SENSe]:WLAN:UNIT:TIME (on page 2481). To set the Frequency units, use [SENSe]:WLAN:UNIT:FREQuency (on page 2480). To set the Radix, use [SENSe]:WLAN:RADix (on page 2471).

NOTE: Use this command only with multicarrier standards and not singlecarrier (802.11b).

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PERRor:WINDow:SELect:PLOT {SYMBols|SCARrier|BOTH}
DISPlay:WLAN:PERRor:WINDow:SELect:PLOT?
```

Arguments

`SYMBols` displays the Symbols graph.

`SCARrier` displays the Subcarriers graph.

`BOTH` displays both graphs.

Examples

`DISP:WLAN:PERR:WIND:SEL:PLOT SCAR` displays the Subcarrier graph on the WLAN Phase Error analysis display.

DISPlay:WLAN:PERRor:WINDow:TRACe:GRATicule:GRID:STATe

Shows or hides the graticule grid on the screen.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PERRor:WINDow:TRACe:GRATicule:GRID:STATe {0|1|OFF|ON}
DISPlay:WLAN:PERRor:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`ON` or 1 specifies to show the graticule.

`OFF` or 0 specifies to turn it off.

Examples

`DISP:WLAN:PERR:WIND:TRAC:GRAT:GRID:STATE ON` specifies to turn the graticule on.

DISPlay:WLAN:PERRor:Y:AUTO (No Query Form)

Automatically selects the vertical scale and position values for the WLAN Phase Error Symbol and Subcarrier plots.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PERRor:Y:AUTO
```

Arguments

None

Examples

`DISP:WLAN:PERR:Y:AUTO` automatically selects the vertical scale and position values for the WLAN Phase Error Symbol and Subcarrier plots.

DISPlay:WLAN:PERRor:Y:OFFSet

Sets or queries the value of the offset from the center (vertical position), in percent terms, for the WLAN Phase Error Symbol and Subcarrier plots. To set the Vertical Scale, use the command `DISPlay:WLAN:PERRor:Y[:SCALe]`.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PERRor:Y:OFFSet <NRf>
DISPlay:WLAN:PERRor:Y:OFFSet?
```

Arguments

Floating point number between limits which are set as a function of the current scale value guaranteed to keep the plot at least partially visible on the display. Positive inputs push the plot down (as viewed by the user) and negative inputs push the plot up. 0 re-centers the plot.

Returns

The value of the offset from the center of the Y axis for the Phase Error plot as a floating point number.

Examples

`DISP:WLAN:PERR:Y:OFFS 150` sets the offset from the center (vertical position) of the Phase Error plot to 150%.

DISPlay:WLAN:PERRor:Y:RESet (No Query Form)

Resets the vertical scale and position (offset) values for the WLAN Phase Error Symbol and Subcarrier plots.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PERRor:Y:RESet
```

Arguments

None

Examples

`DISP:WLAN:PERR:Y:RES` specifies to reset the vertical scale and position (offset) values for the WLAN Phase Error plot.

DISPlay:WLAN:PERRor:Y[:SCALe]

Sets or queries the vertical scale value for the WLAN Phase Error plots. To set the vertical position (offset) value, use the command `DISPlay:WLAN:PERRor:Y:OFFSet`.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PERRor:Y[:SCALe] <NRf>
DISPlay:WLAN:PERRor:Y[:SCALe]?
```

Arguments

Floating point number that specifies the vertical scale value.

Examples

`DISP:WLAN:PERR:Y:SCALE 10` sets the vertical scale value to 10.

DISPlay:WLAN:PVTime:BURSt:X[:SCALe]

Sets or queries the value of the scale (width), in seconds, for the WLAN Power vs. Time display when using horizontal Full Burst view.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PVTime:BURSt:X[:SCALe] <NRf>
DISPlay:WLAN:PVTime:BURSt:X[:SCALe]?
```

Arguments

Floating point number that represents the value of the scale (width), in seconds, when using horizontal Full Burst view.

Examples

`DISP:WLAN:PVT:BURS:X:SCALE 5.0E-6` sets the width of the display to 5.000 µS for the horizontal Full Burst view.

DISPlay:WLAN:PVTime:BURSt:X[:SCALe]:AUTO (No Query Form)

Automatically sets the starting time (position) and scale (width) values for the best display in the horizontal Full Burst view.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PVTime:BURSt:X[:SCALe]:AUTO
```

Arguments

None

Examples

`DISP:WLAN:PVT:BURS:X:SCALE:AUTO` automatically sets the starting time (position) and scale (width) values for the best display.

DISPlay:WLAN:PVTime:BURSt:X[:SCALe]:OFFSet

Sets or queries the starting time (position) value, in seconds, for the WLAN Power vs. Time display when using the horizontal Full Burst view.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PVTime:BURSt:X[:SCALe]:OFFSet <NRf>
DISPlay:WLAN:PVTime:BURSt:X[:SCALe]:OFFSet?
```

Arguments

Floating point number that represents the starting time (position) value, in seconds, when using horizontal Full Burst view.

Examples

`DISP:WLAN:PVT:BURS:X:OFFFSET 5.0E-6` sets the starting time (position) of the graph to 5.000 µS for horizontal Full Burst view.

DISPlay:WLAN:PVTime:BURSt:X[:SCALe]:RESet (No Query Form)

Resets the starting time (position) and scale (width) values for the horizontal Full Burst view on the WLAN Power vs. Time display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PVTime:BURSt:X[:SCALe]:RESet
```

Arguments

None

Examples

`DISP:WLAN:PVT:BURST:X:SCALE:RESET` resets the starting time (position) and scale (width) values for the horizontal Full Burst view.

DISPlay:WLAN:PVTime:FALL:X[:SCALe]

Sets or queries the scale (width) value, in seconds, for the WLAN Power vs. Time display when using the horizontal Falling Edge view.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PVTime:FALL:X[:SCALe] <NRf>
DISPlay:WLAN:PVTime:FALL:X[:SCALe]?
```

Arguments

Floating point number that represents the scale (width), in seconds, when using the horizontal Falling Edge view.

Examples

`DISP:WLAN:PVT:FALL:X:SCALE 5.0E-6` sets the width of the graph to 5.000 µS when using the horizontal Falling Edge view.

DISPlay:WLAN:PVTime:FALL:X[:SCALe]:AUTO (No Query Form)

Automatically sets the starting time (position) and scale (width) values for the best display in the horizontal Falling Edge view.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PVTime:FALL:X[:SCALe]:AUTO
```

Arguments

None

Examples

`DISP:WLAN:PVT:FALL:X:SCALE:AUTO` automatically sets the starting time (position) and scale (width) values for the best display.

DISPlay:WLAN:PVTime:FALL:X[:SCALe]:OFFSet

Sets or queries the value for the starting time (offset) of the graph, in seconds, for the WLAN Power vs. Time display when using the horizontal Falling Edge view.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PVTime:FALL:X[:SCALe]:OFFSet <NRf>
DISPlay:WLAN:PVTime:FALL:X[:SCALe]:OFFSet?
```

Arguments

Floating point number that represents the value of the starting time (offset) of the graph, in seconds, when using the horizontal Falling Edge view.

Examples

`DISP:WLAN:PVT:FALL:X:SCALE 5.0` sets the starting time (position) of the graph to 5.000 µS when using the horizontal Falling Edge view.

DISPlay:WLAN:PVTime:FALL:X[:SCALe]:RESet (No Query Form)

Resets the starting time (position) and scale (width) values for the horizontal Falling Edge view on the WLAN Power vs. Time display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PVTime:FALL:X[:SCALe]:RESet
```

Arguments

None

Examples

`DISP:WLAN:PVT:FALL:X:SCALE:RESET` resets the starting time (position) and scale (width) values for the horizontal Falling Edge view.

DISPlay:WLAN:PVTime:MARKer:SHOW:STATe

Shows or hides the marker readout on the WLAN Power vs. Time display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PVTime:MARKer:SHOW:STATe {0|1|OFF|ON}
DISPlay:WLAN:PVTime:MARKer:SHOW:STATe?
```

Arguments

`ON` or 1 specifies to show the marker readout.

`OFF` or 0 specifies to turn the marker readout off.

Examples

`DISP:WLAN:PVT:MARK:SHOW:STATE ON` specifies to turn the marker readout on.

DISPlay:WLAN:PVTime:RISE:X[:SCALe]

Sets or queries the scale (width) value, in seconds, for the WLAN Power vs. Time display when using the horizontal Rising Edge view.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PVTime:RISE:X[:SCALe] <NRf>
DISPlay:WLAN:PVTime:RISE:X[:SCALe]?
```

Arguments

Floating point number that represents the scale (width) value, in seconds, for the horizontal Rising Edge view.

Examples

`DISP:WLAN:PVT:RISE:X:SCALE 5.0E-6` sets the width of the graph to 5.000µS for the horizontal Rising Edge view.

DISPlay:WLAN:PVTime:RISE:X[:SCALe]:AUTO (No Query Form)

Automatically sets the starting time (position) and scale (width) values for the best display in the horizontal Rising Edge view of the WLAN Power vs. Time display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PVTime:RISE:X[:SCALe]:AUTO
```

Arguments

None

Examples

`DISP:WLAN:PVT:RISE:X:SCALE:AUTO` automatically sets the starting time (position) and scale (width) values for the best display.

DISPlay:WLAN:PVTime:RISE:X[:SCALe]:OFFSet

Sets or queries the starting time value (offset), in seconds, for the WLAN Power vs. Time display when using the horizontal Rising Edge view.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PVTime:RISE:X[:SCALe]:OFFSet <NRf>
DISPlay:WLAN:PVTime:RISE:X[:SCALe]:OFFSet?
```

Arguments

Floating point number that represents the value of the starting time (offset) of the graph, in seconds, for the horizontal Rising Edge view.

Examples

`DISP:WLAN:PVT:RISE:X:OFFSET 5.0e-6` sets the starting time (position) of the graph to 5.000 µS for the horizontal Rising Edge view.

DISPlay:WLAN:PVTime:RISE:X[:SCALe]:RESet (No Query Form)

Resets the starting time (position) and scale (width) values for the horizontal Rising Edge view on the WLAN Power vs. Time display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PVTime:RISE:X[:SCALe]:RESet
```

Arguments

None

Examples

`DISP:WLAN:PVT:RISE:X:SCALE:RESET` resets the starting time (position) and scale (width) values for the horizontal Rising Edge view.

DISPlay:WLAN:PVTime:WINDow:SELect:PLOT

Specifies or queries which view to use for the WLAN Power vs. Time display. Select from Full Burst, Rising Edge or Falling Edge.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PVTime:WINDow:SELect:PLOT {BURSt|RISE|FALL}
DISPlay:WLAN:PVTime:WINDow:SELect:PLOT?
```

Arguments

BURSt sets the view to Full Burst, which displays the entire packet, with vertical lines indicating the length of the burst.

RISE sets the view to Rising Edge, which zooms the display into the interval around the burst rising edge, with vertical lines indicating the 10% to 90% Power-On Ramp time.

FALL sets the view to Falling Edge, which zooms the display into the interval around the burst Falling Edge, with vertical lines indicating the 90% to 10% Power-Down Ramp time.

Returns

DISP:WLAN:PVT:WIND:SEL:PLOT BURS sets the view to Full Burst.

DISP:WLAN:PVT:WIND:SEL:PLOT? might return RISE, indicating that the view has been set to Rising Edge.

DISPlay:WLAN:PVTime:WINDow:TRACe:GRATicule:GRID:STATe

Shows or hides the graticule grid on the screen.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PVTime:WINDow:TRACe:GRATicule:GRID:STATe {0|1|OFF|ON}
DISPlay:WLAN:PVTime:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

ON or 1 specifies to show the graticule.

OFF or 0 specifies to turn it off.

Examples

DISP:WLAN:PVT:WIND:TRAC:GRAT:GRID:STATE ON specifies to turn the graticule on.

DISPlay:WLAN:PVTime:Y[:SCALe]

Sets or queries the vertical scale value for the WLAN Power vs. Time display.

To set the vertical position (offset) value, use the command DISPlay:WLAN:PVTime:Y[:SCALe]:OFFSet (on page 1057).

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PVTime:Y[:SCALe] <NRf>
DISPlay:WLAN:PVTime:Y[:SCALe]?
```

Arguments

Floating point number that specifies the vertical scale value for the WLAN Power vs. Time display.

Examples

`DISP:WLAN:PVT:Y:SCALE 10` sets the vertical scale value to 10 dB.

DISPlay:WLAN:PVTime:Y[:SCALe]:AUTO (No Query Form)

Automatically selects the vertical scale and position values for the WLAN Power vs. Time display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PVTime:Y[:SCALe]:AUTO
```

Arguments

None

Examples

`DISP:WLAN:PVT:Y:AUTO` automatically selects the vertical scale and position values for the display.

DISPlay:WLAN:PVTime:Y[:SCALe]:OFFSet

Sets or queries the value of the vertical offset (top edge of the vertical axis) for the WLAN Power vs. Time display. To set the vertical scale, use the command `DISPlay:WLAN:PVTime:Y[:SCALe]`.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PVTime:Y[:SCALe]:OFFSet <NRf>
DISPlay:WLAN:PVTime:Y[:SCALe]:OFFSet?
```

Arguments

Floating point number between limits which specifies the value of the vertical offset. Positive inputs push the plot down (as viewed by the user) and negative inputs push the plot up.

Returns

The value of the vertical offset for the WLAN Power vs. Time display as a floating point number.

Examples

`DISP:WLAN:PVT:Y:OFFS -10` sets the vertical offset to −10 dBm.

DISPlay:WLAN:PVTime:Y[:SCALe]:PDIVision

Sets or queries the value of the vertical scale for the WLAN Power vs. Time display, in dB/division. This is only a visual control for panning the graph.

Group

Display commands

Syntax

```
DISPlay:WLAN:PVTime:Y[:SCALe]:PDIVision <NRf>
DISPlay:WLAN:PVTime:Y[:SCALe]:PDIVision?
```

Arguments

Floating point value that represents the value of the vertical scale, in dB/division, for the WLAN Power vs. Time display.

Examples

`DISP:WLAN:PVT:Y:SCALE:PDIV 20` sets the vertical scale to 20.0 dB/division.

DISPlay:WLAN:PVTime:Y[:SCALe]:RESet (No Query Form)

Resets the vertical scale and position (offset) values for the WLAN Power vs. Time display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PVTime:Y[:SCALe]:RESet
```

Arguments

None

Examples

`DISP:WLAN:PVT:Y:RES` specifies to reset the vertical scale and position (offset) values for the WLAN Power vs. Time display.

DISPlay:WLAN:SUMMary:EVMUnits

Sets or queries the EVM Units setting in the WLAN Summary display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:SUMMary:EVMUnits {PERCent | DB}
DISPlay:WLAN:SUMMary:EVMUnits?
```

Arguments

`PERCent`. Units in the WLAN Summary display are shown as percentages.

`DB`. Units in the WLAN Summary display are shown in decibels (dB).

*ESE

Sets or queries the bits in the Event Status Enable Register (ESER). The ESER prevents events from being reported to the Status Byte Register (STB). Refer to Status and Events (on page 2740) for the register information.

Conditions

Measurement views: All

Group

IEEE common commands

Syntax

```
*ESE <value>
*ESE?
```

Related Commands

*CLS (on page 664)

*ESR? (on page 1059)

*SRE (on page 2482)

*STB? (on page 2531)

Arguments

`<value>::<NR1>` is a value in the range from 0 through 255.

The binary bits of the ESER are set according to this value.

Examples

`*ESE 145` sets the ESER to binary 10010001, which enables the PON, EXE, and OPC bits.

`*ESE?` might return the string `*ESE 184`, showing that the ESER contains the binary value 10111000.

*ESR? (Query Only)

Returns the contents of the Standard Event Status Register (SESR). `*ESR?` also clears the SESR (since reading the SESR clears it). Refer to Status and Events (on page 2740) for the register information.

Conditions

Measurement views: All

Group

IEEE common commands

Syntax

```
*ESR?
```

Related Commands

*CLS (on page 664)

*ESE (on page 1059)

*SRE (on page 2482)

*STB? (on page 2531)

Arguments

None

Returns

<NR1> representing the contents of the SESR by a 0 to 255 decimal number.

Examples

*ESR? might return the value 213, showing that the SESR contains binary 11010101.

Fetch commands

FETCh:ACPower? (Query Only)

Returns the Channel power and ACPR measurement results for all available channels.

Conditions

Measurement views: Channel power and ACPR

Group

Fetch commands

Syntax

```
FETCh:ACPower?
```

Arguments

None

Returns

<chan_power>,<acpr_lower(1)>,<acpr_upper(1)>,

<acpr_lower(2)>,<acpr_upper(2)>,...

<acpr_lower(n)>,<acpr_upper(n)>

Where

<chan_power> is the average power of the main channel as the power reference in dBm. The unit can be changed by the [SENSe]:POWer:UNITs command.

<acpr_lower(n)> is the ACPR for the lower channel #n in dB.

<acpr_upper(n)> is the ACPR for the upper channel #n in dB.

The number of n depends on the setting of the [SENSe]:ACPower:CHANnel:PAIRs command.

Examples

FETCH:ACPOWER? might return 4.227,-28.420,-23.847,-22.316,-29.225, indicating (average power of the main channel) = 4.227 dBm, (ACPR for the lower channel 1) = -28.420 dB, (ACPR for the upper channel 1) = -23.847 dB, (ACPR for the lower channel 2) = -22.316 dB, and (ACPR for the upper channel 2) = -29.225 dB.

FETCh:ACPower:CHANnel:POWer? (Query Only)

Returns the average power of the main channel (power reference) in the Channel power and ACPR measurement.

Conditions

Measurement views: Channel power and ACPR

Group

Fetch commands

Syntax

FETCh:ACPower:CHANnel:POWer?

Arguments

None

Returns

<chan_power>::<NRf> is the average power of the main channel in dBm.

The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples

FETCH:ACPOWER:CHANNEL:POWER? might return 4.227, indicating that the average power of the main channel is 4.227 dBm.

FETCh:ACPower:SPECtrum? (Query Only)

Returns spectrum trace data of the Channel power and ACPR measurement.

Conditions

Measurement views: Channel power and ACPR

Group

Fetch commands

Syntax

FETCh:ACPower:SPECtrum?

Arguments

None

Returns

#<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the amplitude in dBm at the nth data point,

4-byte little endian floating-point format specified in IEEE 488.2. The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples

FETCH:ACPOWER:SPECTRUM? might return #43204xxxx... (3204-byte data) for the spectrum trace data of the Channel power and ACPR measurement.

FETCh:{AM|FM|PM}? (Query Only)

Returns the trace data in the AM/FM/PM measurement.

Conditions

Measurement views: AM, FM, and PM

Group

Fetch commands

Syntax

FETCh:{AM|FM|PM}?

Arguments

None

Returns

#<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the modulation factor in percent (AM), frequency deviation in Hz (FM), or phase deviation in degrees (PM) at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

FETCH:AM? might return #3156xxxx... (156-byte data) for the AM measurement trace.

FETCh:AM:AMINdex? (Query Only)

Returns the modulation depth which is the (positive peak modulation factor - negative peak modulation factor)/2 returned as a percentage (%).

Conditions

Measurement views: AM

Group

Fetch commands

Syntax

FETCh:AM:AMINdex?

Arguments

None

Returns

<value>::<NRf> the modulation index.

Examples

FETCH:AM:AMINDEX? might return 77.1854035556E-3, indicating the modulation index is 0.0772% or 77.2 m%.

FETCh:AM:AMNegative? (Query Only)

Returns the negative peak modulation factor (–AM) in the AM measurement.

Conditions

Measurement views: AM

Group

Fetch commands

Syntax

FETCh:AM:AMNegative?

Arguments

None

Returns

<-AM>::<NRf> is the negative peak modulation factor in percent (%).

Examples

FETCH:AM:AMNEGATIVE? might return -23.4, indicating the negative peak modulation factor is –23.4%.

FETCh:AM:AMPositive? (Query Only)

Returns the positive peak modulation factor (+AM) in the AM measurement.

Conditions

Measurement views: AM

Group

Fetch commands

Syntax

FETCh:AM:AMPositive?

Arguments

None

Returns

<+AM>::<NRf> is the positive peak modulation factor in percent (%).

Examples

`FETCH:AM:AMPOSITIVE?` might return `43.8`, indicating the positive peak modulation factor is 43.8%.

FETCh:AM:RESult? (Query Only)

Returns the AM measurement results.

Conditions

Measurement views: AM

Group

Fetch commands

Syntax

`FETCh:AM:RESult?`

Arguments

None

Returns

`<+AM>,<-AM>,<Total AM>`

Where

`<+AM>::<NRf>` is the positive peak modulation factor in percent (%).

`<-AM>::<NRf>` is the negative peak modulation factor in percent (%).

`<Total AM>::<NRf>` is the (peak to peak modulation factor)/2 in percent (%).

Examples

`FETCH:AM:RESULT?` might return `62.63,-50.89,56.76`.

FETCh:AUDio:FERRor? (Query Only)

Returns the carrier frequency error in the audio measurement. This command is only available for FM and PM measurements.

Conditions

Measurement views: Audio Summary

Group

Fetch commands

Syntax

`FETCh:AUDio:FERRor?`

Arguments

None

Returns

<value> ::= <NRf> the carrier frequency error.

Examples

`FETCH:AUDIO:FERROR?` might return `419.9529809622` indicating that the audio frequency error is approximately 420 Hz.

FETCh:AUDio:FREQuency? (Query Only)

Returns the audio frequency.

Conditions

Measurement views: Audio Summary

Group

Fetch commands

Syntax

`FETCh:AUDio:FREQuency?`

Arguments

None

Returns

`<value> ::= <NRf>` the audio frequency.

Examples

`FETCH:AUDIO:FREQUENCY?` might return 5.8239462705E+3 indicating that the audio frequency is 5.824 kHz.

FETCh:AUDio:HARMonic:COUNt? (Query Only)

Returns the number of harmonics in the audio spectrum measurement.

The number of harmonics appear in the results table in the audio spectrum view.

Conditions

Measurement views: Audio Spectrum

Group

Fetch commands

Syntax

`FETCh:AUDio:HARMonic:COUNt?`

Related Commands

FETCh:AUDio:NHARmonic:COUNt? (on page 1069)

Arguments

None

Returns

`<count> ::= <NR1>` the number of harmonics in the measurement view.

Examples

`FETCH:AUDIO:HARMONIC:COUNT?` might return 3 indicating that there are three harmonics in the measurement.

FETCh:AUDio:HARMonic<x>:AMPLitude? (Query Only)

Returns the amplitude of the specified harmonic in the audio spectrum measurement.

The value <x> represents the specific harmonic, from 1 to the value specified by the `CALCulate:AUDio:HARMonic:HNUMber` command.

Conditions

Measurement views: Audio Spectrum

Group

Fetch commands

Syntax

`FETCh:AUDio:HARMonic<x>:AMPLitude?`

Related Commands

CALCulate:AUDio:HARMonic:HNUMber (on page 351)

Arguments

None

Returns

`<value> ::= <NRf>` the amplitude of the specified harmonic in dBc.

Examples

`FETCh:AUDio:HARMonic3:AMPLitude?` might return -2.861 indicating that amplitude of the third harmonic is -2.86 dBc.

FETCh:AUDio:HARMonic<x>:FREQuency? (Query Only)

Returns the frequency of the specified harmonic in the audio spectrum measurement.

The value <x> represents the specific harmonic, from 1 to the value specified by the `CALCulate:AUDio:HARMonic:HNUMber` command.

Conditions

Measurement views: Audio Spectrum

Group

Fetch commands

Syntax

`FETCh:AUDio:HARMonic<x>:FREQuency?`

Related Commands

CALCulate:AUDio:HARMonic:HNUMber (on page 351)

Arguments

None

Returns

<value> ::= <NRF> the frequency of the specified harmonic.

Examples

FETCh:AUDio:HARMonic3:FREQuency? might return 17.4718007813E+3, indicating that the frequency of the third harmonic is 17.472 kHz.

FETCh:AUDio:HNOise? (Query Only)

Returns the difference between the current RMS modulation value and the reference value.

This information is only available when the Hum & Noise is measurement is enabled; see the [SENSe]:AUDio:HNOise:ENABle command. The reference value is stored after capturing the reference by pressing the Capture Reference button or by enabling the [SENSe]:AUDio:HNOise:REFerence command.

Conditions

Measurement views: Audio Summary

Group

Fetch commands

Syntax

FETCh:AUDio:HNOise?

Related Commands

[SENSe]:AUDio:HNOise:ENABle (on page 1973)

[SENSe]:AUDio:HNOise:REFerence (on page 1974)

Arguments

None

Returns

<value> ::= <NRF> specifies the difference between the Hum & Noise reference and the actual signal in dB.

Examples

FETCH:AUDIO:HNOISE? might return 0.00 dB, indicating that there is no difference between the Hum & Noise reference and the actual signal.

FETCh:AUDio:HNREFerence? (Query Only)

Returns the Hum and Noise RMS modulation reference value.

The value is stored after capturing the reference by pressing the Capture Reference button or by enabling the [SENSe]:AUDio:HNOise:REFerence command. This information is only available when the Hum & Noise measurement is enabled; see the [SENSe]:AUDio:HNOise:ENABle command.

For AM signal types, the reference represents modulation depth. For FM signal types, the reference represents frequency deviation. For PM signal types the reference represents phase. For Direct signal types the reference represents power.

Conditions

Measurement views: Audio Summary

Group

Fetch commands

Syntax

`FETCh:AUDio:HNREFerence?`

Related Commands

[SENSe]:AUDio:HNOise:ENABle (on page 1973)

[SENSe]:AUDio:HNOise:REFerence (on page 1974)

Arguments

None

Returns

`<value>` ::= <NRf> the Hum & Noise reference frequency.

Examples

`FETCH:AUDIO:HNREFERENCE?` might return `20.581121E+3` indicating that the Hum & Noise reference frequency is 20.58 kHz.

FETCh:AUDio:HPTPeak? (Query Only)

Returns the half peak-to-peak modulation excursion (where the modulation excursion depends on the signal type).

For AM signal types, the modulation excursion is “% Modulation Depth.” For FM signal types, the modulation excursion is “Frequency Deviation.” For PM signal types, the modulation excursion is “Phase Deviation.” For Direct signal types, there is no modulation excursion; it is actually “signal excursion.”

Conditions

Measurement views: Audio Summary

Group

Fetch commands

Syntax

`FETCh:AUDio:HPTPeak?`

Arguments

None

Returns

`<value>` ::= <NRF> specifies the half peak modulation excursion.

Examples

`FETCH:AUDIO:HPTPEAK?` might return `125.28231E+3`, indicating that the half peak modulation excursion is 125.28 kHz.

FETCh:AUDio:MODDist? (Query Only)

Returns the modulation distortion for the audio measurement.

Conditions

Measurement views: Audio Summary

Group

Fetch commands

Syntax

`FETCh:AUDio:MODDist?`

Arguments

None

Returns

<value> ::= <NRf> specifies the modulation distortion.

Examples

`FETCH:AUDIO:MODDIST?` might return 98.6282113 indicating that the modulation distortion is 98.628%.

FETCh:AUDio:NHARmonic:COUNt? (Query Only)

Returns the number of non-harmonics in the audio spectrum measurement.

The number of non-harmonics appear in the results table in the audio spectrum view.

Conditions

Measurement views: Audio Spectrum

Group

Fetch commands

Syntax

`FETCh:AUDio:NHARmonic:COUNt?`

Related Commands

FETCh:AUDio:HARMonic:COUNt? (on page 1065)

Arguments

None

Returns

<count> ::= <NR1> the number of non-harmonics in the measurement view.

Examples

`FETCH:AUDIO:NHARMONIC:COUNT?` might return `7`, indicating that there are seven harmonics in the measurement.

FETCh:AUDio:NHARmonic<x>:AMPLitude? (Query Only)

Returns the amplitude of the specified non-harmonic in the audio spectrum measurement.

The value <x> represents the specific non-harmonic, from 1 to the value specified by the `CALCulate:AUDio:HARMonic:NHNumber` command.

Conditions

Measurement views: Audio Spectrum

Group

Fetch commands

Syntax

`FETCh:AUDio:NHARmonic<x>:AMPLitude?`

Related Commands

CALCulate:AUDio:HARMonic:NHNumber (on page 352)

Arguments

None

Returns

<value> ::= <NRf> the amplitude of the specified non-harmonic in dBc.

Examples

`FETCh:AUDio:NHARMonic3:AMPLitude?` might return `-2.861`, indicating that amplitude of the third non-harmonic is -2.86 dBc.

FETCh:AUDio:NHARmonic<x>:FREQuency? (Query Only)

Returns the frequency of the specified non-harmonic in the audio spectrum measurement.

The value <x> represents the specific non-harmonic, from 1 to the value specified by the `CALCulate:AUDio:HARMonic:NHNumber` command.

Conditions

Measurement views: Audio Spectrum

Group

Fetch commands

Syntax

`FETCh:AUDio:NHARmonic<x>:FREQuency?`

Related Commands

CALCulate:AUDio:HARMonic:NHNumber (on page 352)

Arguments

None

Returns

<value> ::= <NRF> the frequency of the specified non-harmonic.

Examples

`FETCh:AUDio:NHARMonic3:FREQuency?` might return `17.4718007813E+3`, indicating that the frequency of the third non-harmonic is 17.472 kHz.

FETCh:AUDio:PNEGative? (Query Only)

Returns the minus peak modulation excursion (where the modulation excursion depends on the signal type).

For AM signal types, the modulation excursion is “% Modulation Depth.” For FM signal types, the modulation excursion is “Frequency Deviation.” For PM signal types, the modulation excursion is “Phase Deviation.” For Direct signal types, there is no modulation excursion; it is actually “signal excursion.”

Conditions

Measurement views: Audio Summary

Group

Fetch commands

Syntax

`FETCh:AUDio:PNEGative?`

Arguments

None

Returns

<value> ::= <NRf> specifies the minus peak modulation excursion.

Examples

`FETCH:AUDIO:PNEGATIVE?` might return `-196.04321E+3`, indicating that the minus peak modulation excursion is -196.04 kHz.

FETCh:AUDio:POWer? (Query Only)

Returns the carrier power or signal power depending the signal type.

If the signal type is Direct, the returned value is the average power of the input signal. If the signal type is AM, FM, or PM, the returned value is the carrier power; the average power of the carrier signal with the modulation removed.

Conditions

Measurement views: Audio Summary

Group

Fetch commands

Syntax

`FETCh:AUDio:POWer?`

Arguments

None

Returns

<value> ::= <NRf> specifies the carrier or signal power in dBm.

Examples

`FETCH:AUDIO:POWER?` might return `-22.231123`, indicating that the carrier power is -22.23 dBm.

FETCh:AUDio:PPOSitive? (Query Only)

Returns the positive peak modulation excursion (where the modulation excursion depends on the signal type).

For AM signal types, the modulation excursion is “% Modulation Depth.” For FM signal types, the modulation excursion is “Frequency Deviation.” For PM signal types, the modulation excursion is “Phase Deviation.” For Direct signal types, there is no modulation excursion; it is actually “signal excursion.”

Conditions

Measurement views: Audio Summary

Group

Fetch commands

Syntax

`FETCh:AUDio:PPOSitive?`

Arguments

None

Returns

<value> ::= <NRf> specifies the positive peak modulation excursion.

Examples

`FETCH:AUDIO:PPOSITIVE?` might return `215.04321E+3`, indicating that the positive peak modulation excursion is -215.04 kHz.

FETCh:AUDio:RMS? (Query Only)

Returns the RMS modulation excursion for the audio measurement (where the modulation excursion depends on the signal type).

For AM signal types, the modulation excursion is “% Modulation Depth.” For FM signal types, the modulation excursion is “Frequency Deviation.” For PM signal types, the modulation excursion is “Phase Deviation.” For Direct signal types, there is no modulation excursion; it is actually “signal excursion.”

Conditions

Measurement views: Audio Summary

Group

Fetch commands

Syntax

`FETCh:AUDio:RMS?`

Arguments

None

Returns

`<value> ::= <NRf>` specifies the RMS modulation excursion.

Examples

`FETCH:AUDIO:RMS?` might return `20.575039E+3`, indicating that RMS modulation excursion is 20.575 kHz.

FETCh:AUDio:SINad? (Query Only)

Returns the signal-to-noise and distortion for the audio measurement.

Conditions

Measurement views: Audio Summary

Group

Fetch commands

Syntax

`FETCh:AUDio:SINad?`

Arguments

None

Returns

`<value> ::= <NRf>` specifies the RMS modulation excursion.

Examples

`FETCH:AUDIO:SINAD?` might return `176.229024E-3`, indicating that the signal-to-noise and distortion for the audio measurement is 0.18 dB.

FETCh:AUDio:SNOise? (Query Only)

Returns the signal level to noise level with the harmonic distortion and non-harmonic distortion components removed.

Conditions

Measurement views: Audio Summary

Group

Fetch commands

Syntax

`FETCh:AUDio:SNOise?`

Arguments

None

Returns

<value> ::= <NRf> specifies the signal level to noise level.

Examples

FETCH:AUDIO:SNOISE? might return –12.8156364, indicating the signal level to noise level is -12.82 dB.

FETCh:AUDio:SPECtrum:TRACe<x>? (Query Only)

Returns the audio spectrum trace data for the audio measurement.

The <x> represents trace 1 for this measurement. No other traces are supported.

Conditions

Measurement views: Audio Spectrum

Group

Fetch commands

Syntax

FETCh:AUDio:SPECtrum:TRACe<x>?

Arguments

None

Returns

#<num_digit><num_byte><date(1)><data(2)> . . . <data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point in percent (%), 4-byte little endian floating point format specifed in IEEE.488.2.

Examples

FETCh:AUDio:SPECtrum:TRACe1? might return #516004xxxx ... (16004 byte data) for the audio measurement.

FETCh:AUDio:THDist? (Query Only)

Returns the percentage of the total harmonic distortion in the audio measurement.

Conditions

Measurement views: Audio Summary

Group

Fetch commands

Syntax

FETCh:AUDio:THDist?

Related Commands

FETCh:AUDio:THDist:DB? (on page 1075)

Arguments

None

Returns

`<value> ::= <NRf>` specifies the percentage of the total harmonic distortion.

Examples

`FETCH:AUDIO:THDIST?` might return `53.332921`, indicating the percentage of total harmonic distortion is 53.333%.

FETCh:AUDio:THDist:DB? (Query Only)

Returns the total harmonic distortion in dB in the audio measurement.

Conditions

Measurement views: Audio Summary

Group

Fetch commands

Syntax

`FETCh:AUDio:THDist:DB?`

Related Commands

FETCh:AUDio:THDist? (on page 1074)

Arguments

None

Returns

`<value> ::= <NRf>` specifies the total harmonic distortion in dB.

Examples

`FETCH:AUDIO:THDIST:DB?` might return `-5.46009`, indicating that the total harmonic distortion is -5.46 dB.

FETCh:AUDio:TNHDist? (Query Only)

Returns the percentage of the total non-harmonic distortion in the audio measurement.

Conditions

Measurement views: Audio Summary

Group

Fetch commands

Syntax

`FETCh:AUDio:TNHDist?`

Related Commands

FETCh:AUDio:TNHDist:DB? (on page 1076)

Arguments

None

Returns

<value> ::= <NRf> specifies the percentage of the total non-harmonic distortion.

Examples

FETCH:AUDIO:TNHDIST? might return 297.332921, indicating the percentage of total non-harmonic distortion is 297.333%.

FETCh:AUDio:TNHDist:DB? (Query Only)

Returns the total non-harmonic distortion in dB in the audio measurement.

Conditions

Measurement views: Audio Summary

Group

Fetch commands

Syntax

FETCh:AUDio:TNHDist:DB?

Related Commands

FETCh:AUDio:TNHDist? (on page 1075)

Arguments

None

Returns

<value> ::= <NRf> specifies the total non-harmonic distortion in dB.

Examples

FETCH:AUDIO:TNHDIST:DB? might return 9.46009, indicating that the total harmonic distortion is 9.46 dB.

FETCh:AVTime[:MEASview<y>]:AVERage? (Query Only)

Returns the RMS (root-mean-square) value for the selected trace in the Amplitude versus Time measurement. Select the trace using the TRACe<x>:AVTime[:MEASview<y>]:SELect command.

Conditions

Measurement views: Amplitude versus Time

Group

Fetch commands

Syntax

FETCh:AVTime[:MEASview<y>]:AVERage?

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

<avg>::<NRf> is the RMS amplitude in dBm.

The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples

FETCH:AVTIME:MEASVIEW1:AVERAGE? might return -2.53, indicating the RMS amplitude is -2.53 dBm for Amplitude versus Time Measurement View 1.

FETCh:AVTime[:MEASview<y>]:{FIRSt|SECond|THIRd|FOURth}? (Query Only)

Returns the trace data in the Amplitude versus Time measurement.

The mnemonics FIRSt, SECond, THIRd, and FOURth represent Trace 1, Trace 2, Trace 3, and Math trace, respectively. The traces can be specified by the TRACe<x>:AVTime command subgroup.

Conditions

Measurement views: Amplitude versus Time

Group

Fetch commands

Syntax

FETCh:AVTime[:MEASview<y>]:{FIRSt|SECond|THIRd|FOURth}?

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

#<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the amplitude in dBm at the nth data point,

4-byte little endian floating-point format specified in IEEE 488.2. The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples

FETCH:AVTIME:MEASVIEW1:FIRST? might return #3156xxxx... (156-byte data) for Trace 1 for the Amplitude versus Time Measurement View 1.

FETCh:AVTime[:MEASview<y>]:MAXimum? (Query Only)

Returns the maximum value for the selected trace in the Amplitude versus Time measurement. Select the trace using the TRACe<x>:AVTime[:MEASview<y>]:SELect command.

Conditions

Measurement views: Amplitude versus Time

Group

Fetch commands

Syntax

`FETCh:AVTime[:MEASview<y>]:MAXimum?`

Related Commands

FETCh:AVTime[:MEASview<y>]:MAXLocation? (on page 1078)

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

`<max>::<NRf>` is the maximum Amplitude in dBm.

The unit can be changed by the `[SENSe]:POWer:UNITs` command.

Examples

`FETCH:AVTIME:MEASVIEW1:MAXIMUM?` might return `-2.84`, indicating the maximum amplitude is -2.84 dBm for the Amplitude versus Time Measurement View 1.

FETCh:AVTime[:MEASview<y>]:MAXLocation? (Query Only)

Returns the time at which the amplitude is maximum for the selected trace in the Amplitude versus Time measurement. Select the trace using the `TRACe<x>:AVTime[:MEASview<y>]:SELect` command.

Conditions

Measurement views: Amplitude versus Time

Group

Fetch commands

Syntax

`FETCh:AVTime[:MEASview<y>]:MAXLocation?`

Related Commands

FETCh:AVTime[:MEASview<y>]:MAXimum? (on page 1077)

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

`<max_time>::<NRf>` is the time at the maximum in seconds.

Examples

FETCH:AVTIME:MEASVIEW1:MAXLOCATION:FIRST? might return 25.03E-9, indicating the amplitude is maximum at 25.03 ns in Amplitude versus Time Measurement View 1.

FETCh:AVTime[:MEASview<y>]:MINimum? (Query Only)

Returns the minimum value for the selected trace in the Amplitude versus Time measurement. Select the trace using the TRACe<x>:AVTime[:MEASview<y>]:SELect command.

Conditions

Measurement views: Amplitude versus Time

Group

Fetch commands

Syntax

FETCh:AVTime[:MEASview<y>]:MINimum?

Related Commands

FETCh:AVTime[:MEASview<y>]:MINLocation? (on page 1079)

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

<min>::<NRf> is the minimum amplitude in dBm.

The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples

FETCH:AVTIME:MEASVIEW1:MINIMUM? might return -57.64, indicating the minimum amplitude is -57.64 dBm for Amplitude versus Time Measurement View 1.

FETCh:AVTime[:MEASview<y>]:MINLocation? (Query Only)

Returns the time at which the amplitude is minimum for the selected trace in the Amplitude versus Time measurement. Select the trace using the TRACe<x>:AVTime[:MEASview<y>]:SELect command.

Conditions

Measurement views: Amplitude versus Time

Group

Fetch commands

Syntax

FETCh:AVTime[:MEASview<y>]:MINLocation?

Related Commands

FETCh:AVTime[:MEASview<y>]:MINimum? (on page 1079)

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

<min_time>::<NRf> is the time at the minimum in seconds.

Examples

FETCH:AVTIME:MEASVIEW1:MINLOCATION? might return 450.7E-9, indicating the amplitude is minimum at 450.7 ns for Amplitude versus Time Measurement View 1.

FETCh:AVTime[:MEASview<y>]:RESult? (Query Only)

Returns the measurement results for the selected trace in the Amplitude versus Time measurement. Select the trace using the TRACe<x>:AVTime[:MEASview<y>]:SELect command.

Conditions

Measurement views: Amplitude versus Time

Group

Fetch commands

Syntax

FETCh:AVTime[:MEASview<y>]:RESult?

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

<max>,<max_time>,<min>,<min_time>,<rms>

Where

<max>::<NRf> is the maximum amplitude in dBm.

<max_time>::<NRf> is the time at the maximum in seconds.

<min>::<NRf> is the minimum amplitude in dBm.

<min_time>::<NRf> is the time at the minimum in seconds.

<rms>::<NRf> is the RMS amplitude in dBm.

The unit of amplitude can be changed by the [SENSe]:POWer:UNITs command.

Examples

FETCH:AVTIME:MEASVIEW1:RESULT? might return -2.68,48.62E-6,-82.47,22.11E-6,-8.24, indicating that the maximum amplitude is -2.68 dBm at 48.62 µs, the minimum amplitude is -82.47 dBm at 22.11 µs, and the RMS amplitude is -8.24 dBm on Amplitude versus Time Measurement View 1.

FETCh:AVTime[:MEASview<y>]:TRACe:BITMap? (Query Only)

Returns trace waveform data of the bitmap trace in the Amplitude versus Time measurement.

Conditions

Measurement views: Amplitude versus Time

Group

Fetch commands

Syntax

`FETCh:AVTime[:MEASview<y>]:TRACe:BITMap?`

Arguments

None

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1

Returns

A binary block of 32-bit floating point values (IEEE standard single precision floating point format). Each value ranges from 0 to 1.0, where 1.0 represents 100%.

Bitmaps have 201 rows. Data[0] is the first column in the first row, and the following data continues along that row before moving to the next.

Bitmap values are returned in the order of top left pixel first, then the rest of the top row, after that row by row.
In other words, the pixels in the bitmap are returned from left to right and top to bottom; one row at a time.

Examples

`FETCH:AVTIME:TRACE:BITMAP?` might return #6644004xxxx...

:FETCh:AVTime[:MEASview<y>]:AVERage:{FIRSt|SECond|THIRd|FOURth}

Returns the RMS (root-mean-square) value for the selected trace in the Amplitude versus Time measurement. Select the trace using the `:TRACe<x>:AVTime[:MEASview<y>]:SELect` command.

The mnemonics FIRSt, SECond, THIRd, and FOURth represent Trace 1, Trace 2, Trace 3, and Math trace, respectively.

Conditions

Measurement views: Amplitude versus Time

Group

Fetch commands

Syntax

`:FETCh:AVTime[:MEASview<y>]:AVERage:{FIRSt|SECond|THIRd|FOURth}?`

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

`<max>::<NRf>` is the maximum Amplitude in dBm.

The unit can be changed by the `[SENSe]:POWer:UNITs` command

Examples

`:FETCH:AVTIME:MEASVIEW1:AVERAGE:FIRST?` might return -2.53, indicating the RMS amplitude is -2.53 dBm for Trace1 in Amplitude versus Time Measurement View 1.

:FETCh:AVTime[:MEASview<y>]:MAXimum:{FIRSt|SECond|THIRd|FOURth}

Returns the maximum value for the selected trace in the Amplitude versus Time measurement. Select the trace using the `TRACe<x>:AVTime[:MEASview<y>]:SELect` command.

The mnemonics FIRSt, SECond, THIRd, and FOURth represent Trace 1, Trace 2, Trace 3, and Math trace, respectively.

Conditions

Measurement views: Amplitude versus Time

Group

Fetch commands

Syntax

`:FETCh:AVTime[:MEASview<y>]:MAXimum:{FIRSt|SECond|THIRd|FOURth}?`

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

`<max>::<NRf>` is the maximum Amplitude in dBm.

The unit can be changed by the `[SENSe]:POWer:UNITs` command.

Examples

`:FETCH:AVTIME:MEASVIEW1:MAXIMUM:FIRST?` might return -2.84, indicating the maximum amplitude is -2.84 dBm for the Trace1 in Amplitude versus Time Measurement View 1.

:FETCh:AVTime[:MEASview<y>]:MAXLocation:{FIRSt|SECond|THIRd|FOURth}

Returns the time at which the amplitude is maximum for the selected trace in the Amplitude versus Time measurement. Select the trace using the `:TRACe<x>:AVTime[:MEASview<y>]:SELect` command.

The mnemonics FIRSt, SECond, THIRd, and FOURth represent Trace 1, Trace 2, Trace 3, and Math trace, respectively.

Conditions

Measurement views: Amplitude versus Time

Group

Fetch commands

Syntax

`:FETCh:AVTime[:MEASview<y>]:MAXimum:{FIRSt|SECond|THIRd|FOURth}?`

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

<max>::<NRf> is the maximum Amplitude in dBm.

Examples

:FETCH:AVTIME:MEASVIEW1:MAXLOCATION? might return 25.03E-9, indicating the amplitude is maximum at 25.03 ns in Trace1 for Amplitude versus Time Measurement View 1.

:FETCh:AVTime[:MEASview<y>]:MINimum:{FIRSt|SECond|THIRd|FOURth}

Returns the minimum value for the selected trace in the Amplitude versus Time measurement. Select the trace using the :TRACe<x>:AVTime[:MEASview<y>]:SELect command.

The mnemonics FIRSt, SECond, THIRd, and FOURth represent Trace 1, Trace 2, Trace 3, and Math trace, respectively.

Conditions

Measurement views: Amplitude versus Time

Group

Fetch commands

Syntax

:FETCh:AVTime[:MEASview<y>]:MINimum:{FIRSt|SECond|THIRd|FOURth}?

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

<max>::<NRf> is the maximum Amplitude in dBm.

The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples

:FETCH:AVTIME:MEASVIEW1:MINIMUM:FIRST? might return -57.64, indicating the minimum amplitude is -57.64 dBm for Trace1 in Amplitude versus Time Measurement View 1.

FETCh:AVTime[:MEASview<y>]:MINLocation:{FIRSt|SECond|THIRd|FOURth}

Returns the time at which the amplitude is minimum for the selected trace in the Amplitude versus Time measurement. Select the trace using the :TRACe<x>:AVTime[:MEASview<y>]:SELect command.

The mnemonics FIRSt, SECond, THIRd, and FOURth represent Trace 1, Trace 2, Trace 3, and Math trace, respectively.

Conditions

Measurement views: Amplitude versus Time

Group

Fetch commands

Syntax

`:FETCh:AVTime[:MEASview<y>]:MINLocation:{FIRSt|SECond|THIRd|FOURth}?`

Related Commands

:FETCh:AVTime[:MEASview<y>]:MINimum? (on page 1083)

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

`<max>::<NRf>` is the maximum Amplitude in dBm.

Examples

`:FETCH:AVTIME:MEASVIEW1:MINLOCATION:FIRST?` Might return 450.7E-9, indicating the amplitude is minimum at 450.7 ns for Trace1 in Amplitude versus Time Measurement View 1.

FETCh:AVTime[:MEASview<y>]:OGRAm? (Query only)

Returns trace data of a line in the ogram. The line is selected using the `TRACe:AVTime[:MEASview<y>]:OGRAm:SELect:LINE` command.

Conditions

Applicable measurement view must be active.

Group

Fetch commands

Syntax

`FETCh:AVTime[:MEASview<y>]:OGRAm?`

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional. By default y is 1.

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in <num_byte>.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the amplitude in dBm at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2. The unit can be changed by the `[SENSe]:POWer:UNITs` command

Examples

`FETCh:AVTime:MEASview1:OGRAm?` might return #43204xxxx... (3204-byte data) for the line in the ogram.

FETCh:AVTime[:MEASview<y>]:RESult:{FIRSt|SECond|THIRd|FOURth}

Returns the measurement results for the selected trace in the Amplitude versus Time measurement. Select the trace using the `:TRACe<x>:AVTime[:MEASview<y>]:SELect` command.

The mnemonics FIRSt, SECond, THIRd, and FOURth represent Trace 1, Trace 2, Trace 3, and Math trace, respectively.

Conditions

Measurement views: Amplitude versus Time

Group

Fetch commands

Syntax

```
:FETCh:AVTime[:MEASview<y>]:RESult:{FIRSt|SECond|THIRd|FOURth}??
```

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

`<max>,<max_time>,<min>,<min_time>,<rms>`

Where

`<max>::<NRf>` is the maximum amplitude in dBm.

`<max_time>::<NRf>` is the time at the maximum in seconds.

`<min>::<NRf>` is the minimum amplitude in dBm.

`<min_time>::<NRf>` is the time at the minimum in seconds.

`<rms>::<NRf>` is the RMS amplitude in dBm.

The unit of amplitude can be changed by the `[:SENSe]:POWer:UNITs` command.

Examples

`:FETCH:AVTIME:MEASVIEW1:RESULT:FIRST?` might return `-2.68,48.62E-6,-82.47,22.11E-6,-8.24`, indicating that the maximum amplitude is -2.68 dBm at 48.62 µs, the minimum amplitude is -82.47 dBm at 22.11 µs, and the RMS amplitude is -8.24 dBm for Trace1 on Amplitude versus Time Measurement View 1.

FETCh:BIBEmissions:FTX? (Query Only)

Returns the band number corresponding to the transmitted FTX detected in the Bluetooth InBand Emission display.

Conditions

Measurement views: Bluetooth InBand Emission

Group

Fetch commands

Syntax

```
FETCh:BIBEmissions:FTX?
```

Arguments

None

Returns

<value> ::= <NR1> the band number.

Examples

FETCH:BIBEMISSIONS:FTX? might returns 39 indicating the band number corresponding to transmitted FTX.

FETCh:BIBEmissions:POWer? (Query Only)

Returns all channel power results from the Bluetooth InBand Emission display.

Conditions

Measurement views: Bluetooth InBand Emission

Group

Fetch commands

Syntax

FETCh:BIBEmissions:POWer?

Arguments

None

Returns

<Band0_power>,<Band1_power>,<Band2_power>, . . . <Band78_power>

Where:

Bandx_power is the integrated power in that band. All power values are in dBm

Examples

FETCH:BIBEMISSIONS:POWER? might return 4.227,-28.420,-23.847, . . .,-29.225, indicating:

(integrated power for Band 0) = 4.227 dBm,

(Integrated power for Band 1) = -28.420 dBm,

(Integrated power for Band 2) = -23.847 dBm, and

(Integrated power for Band 78) = -29.225 dBm.

FETCh:BIBEmissions:RESUlts:STATus? (Query Only)

Returns the Pass/Fail result status of the Bluetooth InBand Emission display.

Conditions

Measurement views: Bluetooth InBand Emission

Group

Fetch commands

Syntax
`FETCh:BIBEmissions:RESUlts:STATus?`

Arguments
None

Examples
`FETCH:BIBEMISSIONS:RESULTS:STATUS?` returns the Pass/Fail status result on the display.

FETCh:BLUEtooth:CONSte:FERRor? (Query Only)
Returns the frequency error in Hz in the Bluetooth Constellation. The frequency error is the difference between the measured carrier frequency of the signal and the user-selected center frequency.

Conditions
Measurement views: Bluetooth Constellation

Group
Fetch commands

Syntax
`FETCh:BLUEtooth:CONSte:FERRor?`

Arguments
None

Returns
`<freq_error> ::= <NRf>` the frequency error.

Examples
`FETCh:BLUEtooth:CONSte:FERRor?` might return `-10.7E+3`, which is a frequency error of -10.7 kHz.

FETCh:BLUEtooth:CONSte:TRACe? (Query Only)
Queries the Bluetooth Constellation trace data.

Conditions
Measurement views: Bluetooth Constellation

Group
Fetch commands

Syntax
`FETCh:BLUEtooth:CONSte:TRACe?`

Arguments
None

Returns
`#<num_digit><num_byte><I(1)><Q1><I(2)><Q2> . . . <I(n)><Qn>`

Where:

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<I(n)> and <Qn> are the normalized I- and Q-coordinate values at the nth data point. 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

FETCh:BLUEtooth:CONSte:TRACe? might return #43848xxxx . . . (3848-byte data) for the trace.

FETCh:BLUEtooth:EDIagram:FDEViation? (Query Only)

Returns the frequency deviation vs. time trace data with the X values in the Bluetooth eye diagram.

Conditions

Measurement views: Bluetooth eye diagram.

Group

Fetch commands

Syntax

FETCh:BLUEtooth:EDIagram:FDEViation?

Arguments

None

Returns

#<num_digits><num_bytes><Y(1)><X(1)><Y(2)><X(2)> . . . <Y(n)><X(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<Y(1)> and <X(1)> are the normalized I- and Q-coordinate values at the nth datapoint. The 4–byte little endian floating point format is specified in IEEE 488.2.

Examples

FETCh:BLUEtooth:EDIagram:FDEVIiation? might return #3160xxxx... (160-byte data) for the frequency deviation versus time trace.

FETCh:BLUEtooth:EDIagram:FERRor? (Query Only)

Returns the frequency error in the Bluetooth eye diagram measurement.

Conditions

Measurement views: Bluetooth eye diagram.

Group

Fetch commands

Syntax

FETCh:BLUEtooth:EDIagram:FERRor?

Arguments

None

Returns

<freq_error> ::= <NRf> the frequency error in Hz.

Examples

FETCh:BLUEtooth:EDIagram:FERRor? might return -10.7E+3, indicating the frequency error is -10.7 kHz.

FETCh:BLUEtooth:EDIagram:HORiz:SCALe? (Query Only)

Returns the value of the horizontal scale in the Bluetooth eye diagram.

Conditions

Measurement views: Bluetooth eye diagram.

Group

Fetch commands

Syntax

FETCh:BLUEtooth:EDIagram:HORiz:SCALe?

Arguments

None

Returns

<value> ::= <NR3> the horizontal scale value.

Examples

FETCH:BLUETOOTH:EDIAGRAM:HORIZ:SCALE? might return 2, indicating the horizontal scale value is 2 symbols.

FETCh:BLUEtooth:EDIagram:I? (Query Only)

Returns the I vs. time trace in the Bluetooth eye diagram.

Conditions

Measurement views: Bluetooth eye diagram.

Group

Fetch commands

Syntax

FETCh:BLUEtooth:EDIagram:I?

Arguments

None

Returns

#<num_digits><num_bytes><Y(1)><X(1)><Y(2)><X(2)> . . . <Y(n)><X(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<Y(1)> amd <X(1)> are the normalized I- and Q-coordinate values at the nth datapoint. The 4–byte little endian floating point format is specified in IEEE 488.2.

Examples

FETCh:BLUEtooth:EDIagram:I? might return #3160xxxx... (160-byte data) for the I vs. time trace.

FETCh:BLUEtooth:EDIagram:Q? (Query Only)

Returns the Q vs. time trace in the Bluetooth eye diagram.

Conditions

Measurement views: Bluetooth eye diagram.

Group

Fetch commands

Syntax

FETCh:BLUEtooth:EDIagram:Q?

Arguments

None

Returns

#<num_digits><num_bytes><Y(1)><X(1)><Y(2)><X(2)> . . . <Y(n)><X(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<Y(1)> amd <X(1)> are the normalized I- and Q-coordinate values at the nth datapoint. The 4–byte little endian floating point format is specified in IEEE 488.2.

Examples

FETCh:BLUEtooth:EDIagram:Q? might return #3160xxxx... (160-byte data) for the Q vs. time trace.

FETCh:BLUEtooth:FDVTime:TRACe? (Query Only)

Returns the Bluetooth Frequency Deviation vs. Time trace data.

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:FDVTime:TRACe?`

Arguments

None

Returns

`#<num_digits><num_bytes><data(1)><data(2)> . . . <data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the frequency deviation in Hz at the nth datapoint. The 4-byte little endian floating point format is specified in IEEE 488.2.

Examples

`FETCh:BLUEtooth:FDVTime:TRACe?` might return `#3160xxxx...` (160-byte data) for the trace.

FETCh:BLUEtooth:FREQuency:ERROr? (Query Only)

Returns the frequency error from the Bluetooth Constellation diagram.

Conditions

Measurement views: Bluetooth Constellation

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:FREQuency:ERROr?`

Arguments

None

Returns

`<NR3>`, the frequency error value in Hz.

Examples

`FETCH:BLUETOOTH:FREQUENCY:ERROR?` might return `0.0000`, indicating the frequency error is 0.0000 Hz.

FETCh:BLUEtooth:FREQuency:ERROr:TYPE? (Query Only)

Returns the frequency error type in the Bluetooth Constellation display.

Conditions

Measurement views: Bluetooth Constellation

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:FREQuency:ERROr:TYPE?`

Arguments

None

Returns

`1` indicates the error is automatically detected.

`0` indicates the error is not automatically detected.

Examples

`FETCH:BLUETOOTH:FREQUENCY:ERROR:TYPE?` might return `1`, indicating the frequency error is automatically detected.

FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:F1FZero? (Query Only)

Returns the drift in frequency offset (f1–f0 — the frequency offset calculated in the first interval in the payload — the frequency offset calculated in the preamble) in the Bluetooth Frequency Deviation vs. Time display.

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:F1FZero?`

Arguments

None

Returns

`<NR3>` the drift in frequency offset (f1–f0 — the frequency offset calculated in the first interval in the payload — the frequency offset calculated in the preamble)

Examples

`FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:F1FZero?` might return `-207.6465301514`, indicating the drift in frequency offset (f1 to f0 – the frequency offset calculated in the first interval in the payload – the frequency offset calculated in the preamble)

FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFN5? (Query Only)

Returns the maximum drift of the frequency offset in payload intervals spaced 50 µs away in the Bluetooth Frequency Deviation vs. Time display.

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFN5?`

Arguments

None

Returns

`<NR3>` the maximum drift of the frequency offset in payload intervals spaced 50 µs away.

Examples

`FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:FNF5?` might return `-207.6465301514`, indicating the maximum drift of the frequency offset in payload intervals spaced 50 µs away.

FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFN5:INTErval? (Query Only)

Returns the interval at which the maximum drift of frequency offset in a duration of 50 µs occurred.

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFN5:INTErval?`

Arguments

None

Returns

`<NR1>` interval at which the maximum drift of frequency offset in a duration of 50 µs occurred.

Examples

`FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:FNF5:INTErval?` might return `16`, indicating the interval at which the maximum drift of frequency offset in a duration of 50 µs occurred. Here 16 refers to the 16th interval and that the drift of frequency offset calculated in the 16th interval from that calculated in the 11th interval is the maximum among those calculated 50 µs away.

FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFZERO? (Query Only)

Returns the maximum drift of the frequency offset of the intervals in the payload from the preamble packet in the Bluetooth Frequency Deviation vs. Time display.

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFZERO?`

Arguments

None

Returns

`<NR3>` the maximum drift of frequency offset calculated in payload intervals from the offset calculated in the preamble.

Examples

`FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFZero?` might return `-207.6465301514`, indicating that the maximum drift of frequency offset calculated in payload intervals from the offset calculated in the preamble is -207.6465301515 Hz.

FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFZERO:INTErval? (Query Only)

Returns the interval (n) at which the maximum drift fn-f0 occurred in the Bluetooth Frequency Deviation vs. Time display. (The fn-f0 indicates the drift of the frequency offset calculated in each payload interval from the offset calculated in the preamble.)

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFZERO:INTErval?`

Arguments

None

Returns

`<NR1>` the interval at which the maximum drift (fn-f0) occurred.

Examples

`FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFZero:INTErval?` might return `16`, indicating that at the 16th payload interval, the maximum drift occurred. f(16) - f(0) is the maximum drift among all f(n) -f (0).

FETCh:BLUEtooth:FREQuency:OFFSet:MAX? (Query Only)

Returns the maximum frequency offset in the Frequency Deviation vs. Time and the Center Frequency Offset and Drift displays.

Conditions

Measurement views: Frequency Deviation vs. Time, CF Offset

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:FREQuency:OFFSet:MAX?`

Arguments

None

Returns

`<NR3>` the maximum frequency offset value.

Examples

`FETCh:BLUEtooth:FREQuency:OFFSet:MAX?` might return `-189.4632263184`, the frequency offset value.

FETCh:BLUEtooth:FREQuency:OFFSet:MAX:INTErval? (Query Only)

Returns the maximum frequency offset interval in the Frequency Deviation vs. Time and the Center Frequency Offset and Drift displays.

Conditions

Measurement views: Frequency Deviation vs. Time, CF Offset

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:FREQuency:OFFSet:MAX:INTErval?`

Arguments

None

Returns

`<NR1>` the maximum frequency offset interval.

Examples

`FETCh:BLUEtooth:FREQuency:OFFSet:MAX:INTErval?` might return `271`, indicating that the maximum drift f(n) - f(0) occurred at the 271st payload interval.

FETCh:BLUEtooth:FREQuency:OFFSet:PREAmble? (Query Only)

Returns the offset calculated in the preamble region in the Frequency Deviation vs. Time and the Center Frequency Offset and Drift displays.

Conditions

Measurement views: Frequency Deviation vs. Time, CF Offset

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:FREQuency:OFFSet:PREAmble?`

Arguments

None

Returns

`<NR3>` the offset calculated in the preamble region.

Examples

`FETCh:BLUEtooth:FREQuency:OFFSet:PREAmble?` might return `106.4204711914`, indicating the offset calculated in the preamble region.

FETCh:BLUEtooth:MODUlation:CHARacteristics:AVErage:F? (Query Only)

Returns the average frequency value when the selected test pattern is Other in the Bluetooth Frequency Deviation vs. Time display.

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:MODUlation:CHARacteristics:AVErage:F?`

Arguments

None

Returns

`<NRf>` the average frequency.

Examples

`FETCh:BLUEtooth:MODUlation:CHARacteristics:AVErage:F?` might return `140.8309531250E+3`, the average frequency value.

FETCh:BLUEtooth:MODUlation:CHARacteristics:AVErage:F2F1ratio? (Query Only)

Returns the average ratio value for the high and low frequency deviation in the Bluetooth Frequency Deviation vs. Time display.

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Fetch commands

Syntax

```
FETCh:BLUEtooth:MODUlation:CHARacteristics:AVErage:F2F1ratio?
```

Arguments

None

Returns

<NRf> the average ratio value.

Examples

FETCh:BLUEtooth:MODUlation:CHARacteristics:AVErage:F2F1ratio? might return 24.37E-3, the average ratio value.

FETCh:BLUEtooth:MODUlation:CHARacteristics:AVErage:FONE? (Query Only)

Returns the average frequency deviation for the low deviation pattern in the Bluetooth Frequency Deviation vs. Time display.

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Fetch commands

Syntax

```
FETCh:BLUEtooth:MODUlation:CHARacteristics:AVErage:FONE?
```

Arguments

None

Examples

FETCh:BLUEtooth:MODUlation:CHARacteristics:AVErage:FONE? might return 139.8309531250E+3, the average frequency deviation for the low deviation pattern.

FETCh:BLUEtooth:MODUlation:CHARacteristics:AVErage:FTWO? (Query Only)

Returns the average frequency deviation for the high deviation pattern in the Bluetooth Frequency Deviation vs. Time display.

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Fetch commands

Syntax

```
FETCh:BLUEtooth:MODUlation:CHARacteristics:AVErage:FTWO?
```

Arguments

None

Examples

`FETCh:BLUEtooth:MODUlation:CHARacteristics:AVErage:FTWO?` might return `139.8309531250E+3`, the average frequency deviation for the high deviation pattern.

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:COUNt? (Query Only)

Returns the number of packets analyzed to obtain average drift results from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:COUNt?`

Arguments

None

Returns

`<NR1>` the number of packets analyzed.

Examples

`FETCH:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:COUNT?` might return `8`, the number of packets analyzed.

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:F1FZero? (Query Only)

Returns the drift f1–f0 from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:F1FZero?`

Arguments

None

Returns

`<NRf>` the drift f1–f0 value.

Examples

FETCH:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:F1FZERO? might return 2.846E+3, indicating that the drift f1–f0 value is 2.846 kHz.

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:F1FZero:STATus? (Query Only)

Returns the Pass/Fail drift f1–f0 result status in the packet from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax
FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:F1FZero:STATus?

Arguments

None

Returns

<string> the Pass/Fail result status.

Examples

FETCH:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:F1FZERO:STATUS? might return PASS, FAIL, or NA indicating whether the drift f1–f0 is within the specified limit.

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFN5? (Query Only)

Returns the maximum drift of the fn–fn-5 value from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFN5?

Arguments

None

Returns

<NRf> the fn–fn-5 value.

Examples

FETCH:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:FNFN5? might return 2.846E+3 indicating that the maximum drift fn–fn-5 value in the preamble region is -2.846 kHz.

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFN5:STATus? (Query Only)

Returns the Pass/Fail status of the maximum fn–fn-5 drift in the packet from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFN5:STATus?

Arguments

None

Returns

<string> the Pass/Fail result status.

Examples

FETCH:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:FNFN5:STATUS? might return PASS, FAIL, or NA indicating whether the maximum drift fn–fn-5 is within the specified limit.

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFZero? (Query Only)

Returns the maximum drift fn–f0 from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFZero?

Arguments

None

Returns

<NRf> the fn–fn0 value.

Examples

FETCH:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:FNFZERO? might return 4.846E+3, indicating that the drift fn–f0 value in the preamble region is 4.846 kHz.

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFZero:STATus? (Query Only)

Returns the Pass/Fail result status of the fn–f0 maximum drift in the packet from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFZero:STATus?`

Arguments

None

Returns

`<string>` the Pass/Fail result status.

Examples

`FETCH:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:FNFZERO:STATUS?` might return `PASS`, `FAIL`, or `NA` indicating whether the maximum drift fn–f0 is within the specified limit.

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded:CODIng:SCHEme (Query Only)

Returns the offset drift coding scheme value (LE Coded) in Bluetooth Summary display.

Conditions

Measurement view: Bluetooth Summary display

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded:CODIng:SCHEme?`

Arguments

None

Returns

<NR1> 2 or 8

Where 2 is data at 500 Kb/s and 8 is 125 Kb/s

Examples

`FETCH:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:LECODED:CODING:SCHEME` might return `"2"`, indicating the offset drift coding scheme value is 2.

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded:F0FThree (Query Only)

Returns the Drift value f0-f3 for LE Coded in Bluetooth Summary display.

Conditions

Measurement view: Bluetooth Summary display

Group

Fetch commands

Syntax

```
FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded:F0FThree?
```

Arguments

None

Returns

<NRf> Maximum Frequency value in kHz

Examples

FETCH:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:LECODED:F0FTHREE might return 207.2 kHz, indicating the f0-f3 drift value is 207.2 kHz.

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded:F0FThree:STATus (Query Only)

Returns the Drift value Status of f0-f3 for LE Coded in Bluetooth Summary display.

Conditions

Measurement view: Bluetooth Summary display

Group

Fetch commands

Syntax

```
FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded: F0FThree:STATus?
```

Arguments

None

Returns

<String> PASS ,FAIL or NA

PASS indicates that the test passed.

FAIL indicates that the test failed.

NA indicates that the test is not Applicable

Examples

FETCH:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:LECODED:F0FTHREE:STATUS might return Pass, indicating the test result Status of f0-f3 was Pass.

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded:F1FFour (Query Only)

Returns the Drift value of f1-f4 for LE Coded in Bluetooth Summary display.

Conditions

Measurement view: Bluetooth Summary display

Group

Fetch commands

Syntax

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded:F1FFour?

Arguments

None

Returns

<NRf> Maximum Frequency value in kHz

Examples

FETCH:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:LECODED:F1FFOUR might return –10.22 kHz, indicating the f1-f4 drift value is -10.22 kHz.

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded:F1FFour:STATus (Query Only)

Returns the Drift value Status of f1-f4 for LE Coded in Bluetooth Summary display

Conditions

Measurement view: Bluetooth Summary display

Group

Fetch commands

Syntax

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded: F1FFour:STATus?

Arguments

None

Returns

<String> PASS, FAIL or NA

PASS indicates that the test passed.

FAIL indicates that the test failed.

NA indicates that the test is not Applicable

Examples

`FETCH:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:LECODED:F1FFOUR:STATUS?` might return `Pass`, indicating the test result Status of f1-f4 was Pass.

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded:FNFN3 (Query Only)

Returns the Drift value of fn-f(n-3) for LE Coded in Bluetooth Summary display

Conditions

Measurement view: Bluetooth Summary display

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded:FNFN3?`

Arguments

None

Returns

`<NRf>` Maximum Frequency value in kHz

Examples

`FETCH:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:LECODED:FNFN3` might return `-10.42 kHz`, indicating the fn-f(n-3) drift value is -10.42 kHz.

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded:FNFN3:STATus (Query Only)

Returns the Drift value Status of fn-f(n-3) for LE Coded in Bluetooth Summary display.

Conditions

Measurement view: Bluetooth Summary display

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded:FNFN3:STATus?`

Arguments

None

Returns

`<String>` PASS, FAIL or NA

`PASS` indicates that the test passed.

FAIL indicates that the test failed.

NA indicates that the test is not Applicable

Examples

FETCH:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:LECODED:FNFN3:STATUS? might return Pass, indicating the test result Status of fn-f(n-3) was Pass.

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:MAX? (Query Only)

Returns the maximum frequency offset present in the packet from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:MAX?

Arguments

None

Returns

<NRf> the maximum frequency offset value in the packet.

Examples

FETCH:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:MAX? might return 7.446E+3, indicating that the maximum frequency offset value in the packet is 7.446 kHz.

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:MAX:STATus? (Query Only)

Returns the Pass/Fail result status of the frequency offset maximum in the packet from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:MAX:STATus?

Arguments

None

Returns

<string> the Pass/Fail result status.

Examples

FETCH:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:MAX:STATUS? might return PASS, FAIL, or NA indicating whether the maximum frequency offset in the packet is within the specified limit.

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:PREAmble? (Query Only)

Returns the frequency offset value in the preamble region from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:PREAmble?

Arguments

None

Returns

<NRf> the frequency offset value in the preamble region.

Examples

FETCH:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:PREAMBLE? might return 3.546E+3 indicating that the frequency offset value in the preamble region is 3.546 kHz.

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:PREAmble:STATus? (Query Only)

Returns the Pass/Fail result status of the frequency offset in the preamble region.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:PREAmble:STATus?

Arguments

None

Returns

<string> the Pass/Fail result status.

Examples

FETCH:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:PREAMBLE:STATUS? might return PASS, FAIL, or NA indicating whether the frequency offset in the preamble region is within the specified limit.

FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:F:MAX:PERCentage:COUNt? (Query Only)

Returns the number of packets analyzed out of 10 packets for ΔF average from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage: DELTa:F:MAX:PERCentage:COUNt?

Arguments

None

Returns

<NR1> the number of packets analyzed.

Examples

FETCH:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:DELTA:F:MAX:PERCENTAGE:COUNT? might return 3, indicating that three packets were analyzed.

FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:FONE:MAX:PERCentage:COUNt? (Query Only)

Returns the number of packets analyzed out of 10 packets for the ΔF1avg from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:FONE:MAX:PERCentage:COUNt?

Arguments

None

Returns

<NR1> the number of packets analyzed.

Examples

FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:FONE:MAX:PERCentage:COUNt? might return 6, indicating that six packets were analyzed.

FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:FTWO:MAX:PERCentage:COUNt? (Query Only)

Returns the number of packets analyzed out of 10 packets for the ΔF2Max% from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

```
FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage: DELTa:FTWO:MAX:PERCentage:COUNt?
```

Arguments

None

Returns

<NR1> the number of packets analyzed.

Examples

`FETCH:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:DELTA:FTWO:MAX:PERCENTAGE:COUNT?` might return `6`, indicating that six packets were analyzed.

FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:FTWO:MAX:PERCentage? (Query Only)

Returns the percentage of the ΔF2Max values that are above the specified limit.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

```
FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage: DELTa:FTWO:MAX:PERCentage?
```

Arguments

None

Returns

<NRf> the number of packets analyzed.

Examples

`FETCH:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:DELTA:FTWO:MAX:PERCENTAGE?` might return `99.2`, indicating that 99.2% of the deltaF2 maximum values are above the specified limit.

FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:FTWO:MAX:PERCentage:STATus? (Query Only)

Returns the Pass/Fail result status of the ΔF2Max% field from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage: DELTa:FTWO:MAX:PERCentage:STATus?`

Arguments

None

Returns

<string> the Pass/Fail result status.

Examples

`FETCH:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:DELTA:FTWO:MAX:PERCENTAGE:STATUS?` might return `PASS`, `FAIL`, or `NA`, indicating the Pass/Fail status of the ΔF2Max% field in the Bluetooth summary.

FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:F? (Query Only)

Returns the ΔFavg value from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:F?`

Arguments

None

Returns

<NRf> the delta F value.

Examples

`FETCH:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:F?` might return `2.8723E+3`, indicating the delta F average is 2.8723 kHz.

FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:F2F1ratio? (Query Only)

Returns the ratio of the ΔF2avg to ΔF1avg from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

```
FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:F2F1ratio?
```

Arguments

None

Returns

<NRf> the value of the delta F1 to delta F2 ratio.

Examples

`FETCH:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:F2F1RATIO?` might return `225.12E+3`, indicating the average ratio is 225.12 kHz.

FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:F2F1ratio:STATus? (Query Only)

Returns the Pass/Fail results status of the ΔF2avg/ΔF1avg field from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

```
FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage: F2F1ratio:STATus?
```

Arguments

None

Returns

<string> the Pass/Fail result status.

Examples

`FETCH:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:F2F1RATIO:STATUS?` might return `PASS`, `FAIL`, or `NA`, indicating the Pass/Fall result status of the ΔF2avg/ΔF1avg field.

FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:FONE? (Query Only)

Returns the ΔF1avg value from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:FONE?`

Arguments

None

Returns

<NRf> the delta F1 value.

Examples

`FETCH:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:FONE?` might return `155.3672E+3`, indicating the delta F1 average is 155.3672 kHz.

FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:FONE:STATus? (Query Only)

Returns the Pass/Fail results status of the ΔF1avg from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:FONE:STATus?`

Arguments

None

Returns

<string> the Pass/Fail results status.

Examples

`FETCH:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:FONE:STATUS?` might return `PASS`, `FAIL`, or `NA`, indicating whether the delta F1 average is within the limit.

FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:FTWO? (Query Only)

Returns the ΔF2avg value from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:FTWO?`

Arguments

None

Returns

<NRf> the delta F2 value.

Examples

`FETCH:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:FTWO?` might return `225.12E+3`, indicating that the delta F2 value is 225.12 kHz.

FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:INDex (Query Only)

Returns the Modulation index value in Bluetooth Summary display (for LE 1M, LE 2M, LE Coded).

Conditions

Measurement view: Bluetooth Summary display

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:INDex?`

Arguments

None

Returns

<String>

Examples

`FETCH:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:INDEX` might return `"Stable"`, indicating the status of modulation index is “Stable”.

FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:AVErage? (Query Only)

Returns the average output power from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:AVErage?`

Arguments

None

Returns

<NRf> the average output power value in dBm.

Examples

`FETCH:BLUETOOTH:SUMMARY:OUTPUT:POWER:AVERAGE?` might return `-29.1`, indicating that the average output power is -29.1 dBm.

FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:AVErage:STATus? (Query Only)

Returns the Pass/Fail results status of the average output power from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:AVErage:STATus?`

Arguments

None

Returns

<string> the Pass/Fail results status.

Examples

`FETCH:BLUETOOTH:SUMMARY:OUTPUT:POWER:AVERAGE:STATUS?` might return `PASS`, `FAIL`, or `NA`, indicating whether the average output power is within the specified limit.

FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:COUNt? (Query Only)

Returns the number of packets analyzed out of 10 packets for the calculation of the output power from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:COUNt?`

Arguments

None

Returns

<NR1> the number of packets analyzed.

Examples

`FETCH:BLUETOOTH:SUMMARY:OUTPUT:POWER:COUNT?` might return `6`, indicating that six packets were analyzed.

FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:EDR:COUNt? (Query Only)

Returns the number of packets analyzed out of 10 packets for calculation of the output power for EDR from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:EDR:COUNt?`

Arguments

None

Returns

<NR1> the number of packets analyzed.

Examples

`FETCH:BLUETOOTH:SUMMARY:OUTPUT:POWER:EDR:COUNT?` might return `6`, indicating that six packets were analyzed.

FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:EDR:DPSK? (Query Only)

Returns the power in the DPSK portion of the EDR burst from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:EDR:DPSK?`

Arguments

None

Returns

<NRf> the power in the DPSK portion of the EDR burst.

Examples

`FETCH:BLUETOOTH:SUMMARY:OUTPUT:POWER:EDR:DPSK?` might return `-29.2`, indicating that the average output power is -29.2 dBm.

FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:EDR:GPSK? (Query Only)

Returns the power in the GFSK portion of the EDR burst from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:EDR:GPSK?`

Arguments

None

Returns

<NRf> the power in the GFSK portion of the EDR burst.

Examples

`FETCH:BLUETOOTH:SUMMARY:OUTPUT:POWER:EDR:GPSK?` might return `-29.1`, indicating that the average output power is -29.1 dBm.

FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:PEAK? (Query Only)

Returns the peak output power from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:PEAK?`

Arguments

None

Returns

<NRf> the peak output power value in dBm.

Examples

`FETCH:BLUETOOTH:SUMMARY:OUTPUT:POWER:PEAK?` might return `-29.3` indicating that the peak output power is -29.3 dBm.

FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:PEAK:STATus? (Query Only)

Returns the Pass/Fail results status of the peak output power from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:PEAK:STATus?`

Arguments

None

Returns

<string> the Pass/Fail result status.

Examples

`FETCH:BLUETOOTH:SUMMARY:OUTPUT:POWER:PEAK:STATUS?` might return `PASS`, `FAIL`, or `NA`, indicating whether the peak output power is within the specified limit.

FETCh:BLUEtooth:SUMMary:PACKet:BDR:CRC? (Query Only)

Returns the Basic Rate value of the CRC field from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:PACKet:BDR:CRC?`

Arguments

None

Returns

<string> the value of the CRC field.

Examples

`FETCh:BLUEtooth:SUMMary:PACKet:Basic Rate:CRC?` might return `"0x9DB0"` indicating the value of the CRC field in hexadecimal.

FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:ARQN? (Query Only)

Returns the value of ARQN field from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:ARQN?`

Arguments

None

Returns

<NR2> the value of the ARQN field.

Examples

`FETCh:BLUEtooth:SUMMary:PACKet:Basic Rate:HEADer:ARQN?` might return 0.000000 indicating the value of the flow field.

FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:FLOW? (Query Only)

Returns the value of Flow field from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:FLOW?`

Arguments

None

Returns

<NR2> the flow value.

Examples

`FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:FLOW?` might return 1.000000 indicating the value of the flow field.

FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:HEC? (Query Only)

Returns the value of HEC field from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:HEC?`

Arguments

None

Returns

<string> the value of the HEC field.

Examples

`FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:HEC?` might return `"00110010"` indicating the value of the HEC field in binary.

FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:LT:ADDRess? (Query Only)

Returns the LT address from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:LT:ADDRess?`

Arguments

None

Returns

<string> the LT address.

Examples

`FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:LT:ADDR?` might return `"101"` indicating the LT address in binary.

FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:SEQN? (Query Only)

Returns the value of SEQN field from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:SEQN?`

Arguments

None

Returns

<NR2> the value of the SEQN field.

Examples

`FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:SEQN?` might return 1.000000 indicating the value of the flow field.

FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:TYPE? (Query Only)

Returns the value of Type field from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:TYPE?`

Arguments

None

Returns

<string> the value of the type field.

Examples

`FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:TYPE?` might return `"00001111"` indicating the type field in binary.

FETCh:BLUEtooth:SUMMary:PACKet:BDR:PAYLoad:LENgth? (Query Only)

Returns the Basic Rate payload length from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:PACKet:BDR:PAYLoad:LENgth?`

Arguments

None

Returns

<string> the value of the payload length.

Examples

`FETCh:BLUEtooth:SUMMary:PACKet:BDR:PAYLoad:LENgth?` might return `"0101010011"` indicating the payload length value in binary.

FETCh:BLUEtooth:SUMMary:PACKet:BDR:PREAmble? (Query Only)

Returns the preamble of the Basic Rate packet from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

FETCh:BLUEtooth:SUMMary:PACKet:BDR:PREAmble?

Arguments

None

Returns

<string> the preamble of the packet.

Examples

FETCh:BLUEtooth:SUMMary:PACKet:BDR:PREAmble? might return "0101" indicating the preamble is 0101 in binary.

FETCh:BLUEtooth:SUMMary:PACKet:BDR:SYNC:WORD? (Query Only)

Returns the sync word from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

FETCh:BLUEtooth:SUMMary:PACKet:BDR:SYNC:WORD?

Arguments

None

Returns

<string> the sync word.

Examples

FETCh:BLUEtooth:SUMMary:PACKet:BDR:SYNC:WORD? might return "Ox4F36F2CEE85390CB" indicating the sync word in hexadecimal format.

FETCh:BLUEtooth:SUMMary:PACKet:BDR:TYPE? (Query Only)

Returns the packet type of the Basic Rate signal from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

FETCh:BLUEtooth:SUMMary:PACKet:BDR:TYPE?

Arguments

None

Returns

<string> the packet type.

Examples

`FETCh:BLUEtooth:SUMMary:PACKet:BDR:TYPE?` might return `"DH5"` indicating that the DH5 packet is being analyzed.

FETCh:BLUEtooth:SUMMary:PACKet:BLE:ACCEss:ADDRess? (Query Only)

Returns the 32-bit Access Address from the Bluetooth Summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:PACKet:BLE:ACCEss:ADDRess?`

Arguments

None

Returns

<string> the 32-bit Access Address.

Examples

`FETCH:BLUETOOTH:SUMMARY:PACKET:BLE:ACCESS:ADDRESS?` might return `"0x71764129"` indicating the Access Address in hexadecimal.

FETCh:BLUEtooth:SUMMary:PACKet:BLE:CRC? (Query Only)

Returns the value of the CRC field from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:PACKet:BLE:CRC?`

Arguments

None

Returns

<string> the value of the CRC field.

Examples

`FETCH:BLUETOOTH:SUMMARY:PACKET:BLE:CRC?` might return `"0xAAAAAA"` indicating value CRC field in hexadecimal.

FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:CHANnel:SELect (Query Only)

Returns the channel selected under decoded packet information section in Bluetooth Summary display.

Conditions

Measurement view: Bluetooth Summary display

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:CHANnel:SELect?`

Arguments

None

Returns

<String> 0 or 1 or -1 or "–"

0 and 1 represents valid

-1 represents invalid

"–" represents NIL

Examples

`FETCH:BLUETOOTH:SUMMARY:PACKET:BLE:PDU:HEADER:CHANNEL:SELECT` might return `"1"`, indicating the selected channel is valid.

FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:LENGth? (Query Only)

Returns the value of the length field from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:LENGth?`

Arguments

None

Returns

<string> the value of the length field.

Examples

`FETCH:BLUETOOTH:SUMMARY:PACKET:BLE:PDU:HEADER:LENGTH?` might return `"100101"` indicating value length field in binary.

FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:LLID (Query Only)

Returns the LLID value (2 bits) under decoded packet information when the packet type is “OTHERS” for Low Energy in Bluetooth Summary display.

Conditions

Measurement view: Bluetooth Summary display

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:LLID?`

Arguments

None

Returns

<String> Either 00,01,10,11

Examples

`FETCH:BLUETOOTH:SUMMARY:PACKET:BLE:PDU:HEADER:LLID` might return 01, indicating the value of LLID is "01".

FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:MD (Query Only)

Returns the MD value (1 bit) under decoded packet information when the packet type is “OTHERS” for Low Energy in Bluetooth Summary display.

Conditions

Measurement view: Bluetooth Summary display

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:MD?`

Arguments

None

Returns

<Boolean> 0, 1

Examples

`FETCH:BLUETOOTH:SUMMARY:PACKET:BLE:PDU:HEADER:MD` might return 1, indicating the value of MD is "1".

FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:NESN (Query Only)

Returns the NESN value (1 bit) under decoded packet information when the packet type is “OTHERS” for Low Energy in Bluetooth Summary display.

Conditions

Measurement view: Bluetooth Summary display

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:NESN?`

Arguments

None

Returns

<Boolean> 0, 1

Examples

`FETCH:BLUETOOTH:SUMMARY:PACKET:BLE:PDU:HEADER:NESN` might return 1, indicating the value of NESN is "1".

FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:RX:ADDRess? (Query Only)

Returns the Rx address from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:RX:ADDRess?`

Arguments

None

Returns

<string> the Rx address.

Examples

FETCH:BLUETOOTH:SUMMARY:PACKET:BLE:PDU:HEADER:RX:ADDRESS? might return 0.00000000 indicating the Rx Address.

FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:SN (Query Only)

Returns the SN value (1 bit) under decoded packet information when the packet type is “OTHERS” for Low Energy in Bluetooth Summary display.

Conditions

Measurement view: Bluetooth Summary display

Group

Fetch commands

Syntax

FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:SN?

Arguments

None

Returns

<Boolean> 0, 1

Examples

FETCH:BLUETOOTH:SUMMARY:PACKET:BLE:PDU:HEADER:SN might return 1, indicating the value of SN is "1".

FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:TX:ADDRess? (Query Only)

Returns the Tx address from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:TX:ADDRess?

Arguments

None

Returns

<string> the Tx address.

Examples

`FETCH:BLUETOOTH:SUMMARY:PACKET:BLE:PDU:HEADER:TX:ADDRESS?` might return 0.00000000 indicating Tx Address.

FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:TYPE? (Query Only)

Returns the PDU type from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:TYPE?`

Arguments

None

Returns

<string> the PDU type.

Examples

`FETCH:BLUETOOTH:SUMMARY:PACKET:BLE:PDU:HEADER:TYPE?` might return "0010" indicating the PDU type in binary.

FETCh:BLUEtooth:SUMMary:PACKet:BLE:PREAmble? (Query Only)

Returns the preamble of the BLE packet from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

`FETCh:BLUEtooth:SUMMary:PACKet:BLE:PREAmble?`

Arguments

None

Returns

<string> the preamble of the packet.

Examples

`FETCH:BLUETOOTH:SUMMARY:PACKET:BLE:PREAMBLE?` might return `"01010101"` indicating the preamble is 01010101 in binary.

FETCh:BLUEtooth:SUMMary:PACKet:BLE:TYPE? (Query Only)

Returns the packet type from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Fetch commands

Syntax

```
FETCh:BLUEtooth:SUMMary:PACKet:BLE:TYPE?
```

Arguments

None

Returns

<string> the BLE packet type.

Examples

`FETCH:BLUETOOTH:SUMMARY:PACKET:BLE:TYPE?` might return `"BLE_TEST"` indicating the BLE test signal is being analyzed.

FETCh:CCDF? (Query Only)

Returns the CCDF measurement results.

Conditions

Measurement views: CCDF

Group

Fetch commands

Syntax

```
FETCh:CCDF?
```

Related Commands

Arguments

None

Returns

```
<avg_ampl>,<avg_ccdf>,<crest_factor>,<ampl_10>,<ampl_1>,
   <ampl_p1>,<ampl_p01>,<ampl_p001>,<ampl_p0001>
```

Where

`<avg_ampl>` is the average amplitude in dBm.

The unit can be changed by the `[SENSe]:POWer:UNITs` command.

`<avg_ccdf>` is the average CCDF in percent.

`<crest_factor>` is the crest factor in dB.

<ampl_10> is the amplitude at CCDF of 10% in dB.

<ampl_1> is the amplitude at CCDF of 1% in dB.

<ampl_p1> is the amplitude at CCDF of 0.1% in dB.

<ampl_p01> is the amplitude at CCDF of 0.01% in dB.

<ampl_p001> is the amplitude at CCDF of 0.001% in dB.

<ampl_p0001> is the amplitude at CCDF of 0.0001% in dB.

Examples

FETCH:CCDF? might return -33.35,35.8,9.75,3.88,7.07,8.50,9.25,9.72,9.74, indicating (average amplitude) = -33.35 dBm, (average CCDF) = 35.8%, (crest factor) = 9.75 dB, (amplitude at CCDF of 10%) = 3.88 dB, (amplitude at CCDF of 1%) = 7.07 dB, (amplitude at CCDF of 0.1%) = 8.50 dB, (amplitude at CCDF of 0.01%) = 9.25 dB, (amplitude at CCDF of 0.001%) = 9.72 dB, and (amplitude at CCDF of 0.0001%) = 9.74 dB.

FETCh:CCDF:{FIRSt|SECond|THIRd}:X? (Query Only)

Returns the horizontal values of the specified trace in the CCDF measurement.

The mnemonics FIRSt, SECond, and THIRd represent Trace 1, Trace 2, and Gaussian reference curve, respectively.

Conditions

Measurement views: CCDF

Group

Fetch commands

Syntax

FETCh:CCDF:{FIRSt|SECond|THIRd}:X?

Arguments

None

Returns

#<num_digit><num_byte><x(1)><x(2)>...<x(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<x(n)> is the horizontal value (dB) of the CCDF graph at the nth point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

FETCH:CCDF:FIRST:X? might return #41024xxxx... (1024-byte data) for the horizontal values of Trace 1.

FETCh:CCDF:{FIRSt|SECond|THIRd}:XY? (Query Only)

Returns the horizontal and vertical value pairs of the specified trace in the CCDF measurement.

The mnemonics FIRSt, SECond, and THIRd represent Trace 1, Trace 2, and Gaussian reference curve, respectively.

Conditions

Measurement views: CCDF

Group

Fetch commands

Syntax

```
FETCh:CCDF:{FIRSt|SECond|THIRd}:XY?
```

Arguments

None

Returns

```
#<num_digit><num_byte><x(1)><y(1)><x(2)><y(2)>...<x(n)><y(n)>
```

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<x(n)><y(n)> is the horizontal value (dB) and vertical value (%) pair at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`FETCH:CCDF:FIRST:XY?` might return `#41024xxxx...` (1024-byte data) for the horizontal and vertical value pairs of Trace 1.

FETCh:CCDF:{FIRSt|SECond|THIRd}[:Y]? (Query Only)

Returns the vertical values of the specified trace in the CCDF measurement.

The mnemonics FIRSt, SECond, and THIRd represent Trace 1, Trace 2, and Gaussian reference curve, respectively.

Conditions

Measurement views: CCDF

Group

Fetch commands

Syntax

```
FETCh:CCDF:{FIRSt|SECond|THIRd}[:Y]?
```

Arguments

None

Returns

```
#<num_digit><num_byte><y(1)><y(2)>...<y(n)>
```

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<y(n)> is the vertical value (%) of the CCDF graph at the nth point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`FETCH:CCDF:FIRST:Y?` might return `#41024xxxx...` (1024-byte data) for the vertical values of Trace 1.

FETCh:CONSte[:MEASview<y>]:FERRor? (Query only)

Returns the frequency error in Hz. The frequency error is the difference between the measured carrier frequency of the signal and the user-selected center frequency of the analyzer.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Constellation

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Constellation).

Group

Fetch commands

Syntax

`FETCh:CONSte[:MEASview<y>]:FERRor?`

Related Commands

FETCh:EVM[:MEASview<y>]:FERRor? (on page 1154)

Arguments

None.

Returns

`<freq_error>::<NRf>` which is the frequency error in Hz.

Examples

`FETCh:CONSte:MEASview2:FERRor?` might return -10.7E+3, which is a frequency error of -10.7 kHz for measview 2 of constellation.

FETCh:CONSte[:MEASview<y>]:RESults? (Query only)

Returns the constellation measurement results of **EVM** RMS, peak, and location displayed on the bottom of the screen.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Constellation

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Constellation).

Group

Fetch commands

Syntax

FETCh:CONSte[:MEASview<y>]:RESults?

Arguments

None

Returns

For modulation types 2|4|8|16FSK or C4FM:

FSK_deviation_Avg_Leftmost, FSK_deviation_Avg_Rightmost

Where

FSK_deviation_Avg_Leftmost is the average FSK deviation of the left-most symbol in Hz.

FSK_deviation_Avg_Rightmost is the average FSK deviation of the right-most symbol in Hz.

For all other valid modulation types:

<EVM_RMS>,<EVM_peak>,<location>

Where

<EVM_RMS>::<NRf> is the RMS EVM in percent (%).

<EVM_peak>::<NRf> is the peak EVM in percent (%).

<location>::<NRf> is the peak location in symbol number.

The time unit can be changed by the [SENSe]:DDEMod:TIMe:UNITs command.

Examples

FETCh:CONSTe:MEASview2:RESults? might return 2.841, 3.227, 68.000, indicating that the RMS EVM is 2.841% and the peak EVM is 3.227% at symbol #68 for measview 2 of constellation.

FETCh:CONSte[:MEASview<y>]:TRACe? (Query only)

Returns the constellation trace data.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Constellation

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Constellation).

Group

Fetch commands

Syntax

FETCh:CONSte[:MEASview<y>]:TRACe?

Arguments

None

Returns

#<num_digit><num_byte><I(1)><Q(1)><I(2)><Q(2)>...<I(n)><Q(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<I(n)> and <Q(n)> are the normalized I- and Q-coordinate values at the nth data point. 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

FETCh:CONSte:MEASview2:TRACe? might return #43848xxxx... (3848-byte data) for the constellation trace data for measview 2 of constellation.

FETCh:DDEMod[:MEASview<y>]:STABle? (Query only)

Returns the symbol table data.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Symbol table

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Symbol table)

Group

Fetch commands

Syntax

FETCh:DDEMod[:MEASview<y>]:STABle?

Arguments

None

Returns

#<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the symbol table data at the nth data point,

4-byte little endian floating-point format specified in IEEE 488.2.

Examples

FETCh:DDEMod:MEASview1:STABle? might return #3512xxxx... (512-byte data) for the Symbol table measurement view1.

FETCh:DDEMod:SYNCh:WORD:LENGth? (Query only)

Returns the length of the synch word in the symbol table.

Conditions

Measurement views: Symbol table

Group

Fetch commands

Syntax

`FETCh:DDEMod:SYNCh:WORD:LENGth?`

Related Commands

FETCh:DDEMod:SYNCh:WORD:POSition? (on page 1134)

Arguments

None

Returns

`<NR1>` indicates the length of the synch word in symbols.

Examples

`FETCH:DDEMOD:SYNCH:WORD:LENGTH?` might return 3, indicating the length of the synch word is three symbols.

FETCh:DDEMod:SYNCh:WORD:POSition? (Query Only)

Returns the position of the synch word in the symbol table.

Conditions

Measurement views: Symbol table

Group

Fetch commands

Syntax

`FETCh:DDEMod:SYNCh:WORD:POSition?`

Related Commands

FETCh:DDEMod:SYNCh:WORD:LENGth? (on page 1133)

Arguments

None

Returns

<NR1> indicates what symbol number the synch word begins at in the table. Zero (0) represents the first symbol in the table. A "-1" indicates that the synch word was not found.

Examples

`FETCH:DDEMOD:SYNCH:WORD:POSITION?` might return 10, indicating the synch word begins at 11th symbol in the table.

FETCh:DIQVtime[:MEASview<y>]:FERRor? (Query only)

Returns the frequency error in the Demod I&Q versus Time measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Demod I&Q versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod I&Q versus Time).

Group

Fetch commands

Syntax

`FETCh:DIQVtime[:MEASview<y>]:FERRor?`

Arguments

None

Returns

`<freq_error>::<NRf>` is the frequency error in Hz.

Examples

`FETCh:DIQVtime:MEASview1:FERRor?` might return -10.7E+3, indicating the frequency error is -10.7 kHz measurement view 1.

FETCh:DIQVtime[:MEASview<y>]:I? (Query only)

Returns the I versus Time trace data.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Demod I&Q versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod I&Q versus Time).

Group

Fetch commands

Syntax

`FETCh:DIQVtime[:MEASview<y>]:I?`

Arguments

None

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the I level in volts at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

FETCh:DIQVtime:MEASview1:I? might return #3160xxxx... (160-byte data) for the I versus Time trace measurement view 1.

FETCh:DIQVtime[:MEASview<y>]:Q? (Query only)

Returns the Q versus Time trace data.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Demod I&Q versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod I&Q versus Time).

Group

Fetch commands

Syntax

FETCh:DIQVtime[:MEASview<y>]:Q?

Arguments

None

Returns

#<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the Q level in volts at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

FETCh:DIQVtime:MEASview1:Q? might return #3160xxxx... (160-byte data) for the Q versus Time trace measurement view 1.

FETCh:DPX:DDENsity? (Query Only)

Returns the calculated density for the selected region in the DPXogram measurement.

Conditions

Measurement views: DPX spectrum

Group

Fetch commands

Syntax

FETCh:DPX:DDENsity?

Arguments

None

Returns

<NR1> value.

Examples

`FETCH:DPX:DDENSITY?` might return a DPX signal density value of 24.4802413393.

FETCh:DPX:DGRam:LINE:COUNt? (Query Only)

Returns the number of lines in the DPXogram measurement.

Conditions

Measurement views: DPX spectrum

Group

Fetch commands

Syntax

`FETCh:DPX:DGRam:LINE:COUNt?`

Arguments

None

Returns

<NR1> number of lines in the measurement.

Examples

`FETCH:DPX:DGRAM:LINE:COUNT?` might return `40`, indicating that there were 40 lines in the measurement.

FETCh:DPX:DGRam:TIME[:SCALe]:OFFSet? (Query Only)

Returns the time scale offset in the DPXogram measurement.

Conditions

Measurement views: DPX spectrum

Group

Fetch commands

Syntax

`FETCh:DPX:DGRam:TIME[:SCALe]:OFFSet?`

Arguments

None

Returns

<string> The time offset in the DPXogram measurement in seconds.

Examples

FETCH:DPX:DGRAM:TIME[:SCALE]:OFFSET? might return "0.000 s" indicating that the time scale offset was 0.000 s.

FETCh:DPX:RESults:TRACe<x>? (Query Only)

Returns waveform data of specified trace <x> in the DPX spectrum measurement, where x is 1 to 5. The traces 1–4 are in the standard form. Trace 5 is the bitmap trace and its data is returned in a binary block. Trace 6 is the DPXogram trace on DPX spectrum plots. Trace 7 is the Ogram line in the DPXogram display.

Conditions

Measurement views: DPX spectrum

Group

Fetch commands

Syntax

FETCh:DPX:RESults:TRACe<x>?

Arguments

<NR1>

Returns

For traces 1 to 4:

#<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the amplitude of the trace sample at the nth data point.

<data(n)> is in the 4-byte little endian floating-point format specified in IEEE 488.2.

The format is a binary block of 32 bit floating point values. Each value ranges from 0 to 1.0, where 1.0 represents 100%.

Examples

FETCH:DPX:RESULTS:TRACE1? might return #42004xxxx... (2004-byte of data) for the waveform data of trace one (1).

FETCh:DPX:TDM:FREQuency:TXBer:BER? (Query Only)

Returns the Tx BER measurement BER results.

Conditions

Measurement views: DPX:Frequency

Group

Fetch commands

Syntax

FETCh:DPX:TDM:FREQuency:TXBer:BER?

Related Commands

Arguments

None

Returns

<NR1> is the calculated BER.

Examples

`FETCH:DPX:TDM:FREQUENCY:TXBER:BER?` might return `9.1135684130E-6`.

FETCh:DPX:TDM:FREQuency:TXBer:BITS? (Query Only)

Returns the number of bits counted in the active Tx BER measurement session.

Conditions

Measurement views: DPX:Frequency

Group

Fetch commands

Syntax

`FETCh:DPX:TDM:FREQuency:TXBer:BITS?`

Related Commands

Arguments

None

Returns

<NR1> is the number of bits counted in the current Tx BER measurement session.

Examples

`FETCH:DPX:TDM:FREQUENCY:TXBER:BITS?` might return `72.0820000000E+12`.

FETCh:DPX:TDM:FREQuency:TXBer:ERRors? (Query Only)

Returns the number of errors counted in the active Tx BER measurement session.

Conditions

Measurement views: DPX:Frequency

Group

Fetch commands

Syntax

`FETCh:DPX:TDM:FREQuency:TXBer:ERRors?`

Arguments

None

Returns

<NR1> is the number of errors counted in the current Tx BER measurement session.

Examples

`FETCH:DPX:TDM:FREQUENCY:TXBER:ERRORS?` might return `1.0000000000`.

FETCh:DPX:TDM:FREQuency:TXBer:STATe? (Query Only)

Returns the state of the Tx BER measurement.

Conditions

Measurement views: DPX:Frequency

Group

Fetch commands

Syntax

`FETCh:DPX:TDM:FREQuency:TXBer:STATe?`

Returns

`IDLE` indicates the Tx BER measurement is not running.

`ACTIVE` indicates the Tx BER measurement is in progress.

`SYNC` indicates the instrument is in the process of syncing with the data pattern.

`DONE` indicates the Tx BER measurement is finished.

Examples

`FETCH:DPX:TDM:FREQUENCY:TXBER:STATE?` might return `ACTIVE`, indicating the instrument is measuring the Tx BER.

FETCh:DPX:TRACe:AVERage? (Query Only)

Returns waveform data of the average trace in the DPX spectrum measurement.

Conditions

Measurement views: DPX spectrum

Group

Fetch commands

Syntax

`FETCh:DPX:TRACe:AVERage?`

Arguments

None

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the amplitude of the average trace at the nth data point in dBm, 4-byte little endian floating-point format specified in IEEE 488.2.

The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples

FETCH:DPX:TRACE:AVERAGE? might return #42004xxxx... (2004-byte data) for the waveform data of the average trace.

FETCh:DPX:TRACe:BITMap? (Query Only)

Returns a data block of the DPX spectrum measurement bitmap waveform data.

Conditions

Measurement views: DPX spectrum

Group

Fetch commands

Syntax

FETCh:DPX:TRACe:BITMap?

Arguments

None

Returns

A binary block of 32-bit floating point values (IEEE standard single precision floating point format). Each value ranges from 0 to 1.0, where 1.0 represents 100%. Bitmaps have 201 rows and 801 columns. Data[0] is the first column in the first row, data[801] is the first column in the 2nd row, and so on.

Bitmap values are returned in the following order: Top left pixel first, then the rest of the top row, after that row by row. In other words, the pixels in the bitmap are returned from left to right and top to bottom; one row at a time.

Examples

FETCH:DPX:TRACE:BITMAP? might return #6644004xxxx...

FETCh:DPX:TRACe:MATH? (Query Only)

Returns waveform data of the math trace in the DPX spectrum measurement.

Conditions

Measurement views: DPX spectrum

Group

Fetch commands

Syntax

FETCh:DPX:TRACe:MATH?

Arguments

None

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the amplitude of the math trace at the nth data point in dBm, 4-byte little endian floating-point format specified in IEEE 488.2.

The unit can be changed by the `[SENSe]:POWer:UNITs` command.

Examples

`FETCH:DPX:TRACE:MATH?` might return `#42004xxxx...` (2004-byte data) for the waveform data of the math trace.

FETCh:DPX:TRACe:MAXimum? (Query Only)

Returns waveform data of the maximum trace in the DPX spectrum measurement.

Conditions

Measurement views: DPX spectrum

Group

Fetch commands

Syntax

`FETCh:DPX:TRACe:MAXimum?`

Arguments

None

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the amplitude of the maximum trace at the nth data point in dBm, 4-byte little endian floating-point format specified in IEEE 488.2.

The unit can be changed by the `[SENSe]:POWer:UNITs` command.

Examples

`FETCH:DPX:TRACE:MAXIMUM?` might return `#42004xxxx...` (2004-byte data) for the waveform data of the maximum trace.

FETCh:DPX:TRACe:MINimum? (Query Only)

Returns waveform data of the minimum trace in the DPX spectrum measurement.

Conditions

Measurement views: DPX spectrum

Group

Fetch commands

Syntax

`FETCh:DPX:TRACe:MINimum?`

Arguments

None

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the amplitude data of the minimum trace at the nth data point in dBm, 4-byte little endian floating-point format specified in IEEE 488.2. The unit can be changed by the `[SENSe]:POWer:UNITs` command.

Examples

`FETCH:DPX:TRACE:MINIMUM?` might return `#42004xxxx...` (2004-byte data) for the waveform data of the minimum trace.

FETCh:EDIagram[:MEASview<y>]:FDEViation? (Query only)

Returns the frequency deviation versus Time trace data with the X values.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Eye diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Eye diagram).

Group

Fetch commands

Syntax

`FETCh:EDIagram[:MEASview<y>]:FDEViation?`

Returns

`#<num_digit><num_byte><Y(1)><X(1)><Y(2)><X(2)>...<Y(n)><X(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<Y(n)>` and `<X(n)>` is the frequency deviation in Hz and time (symbols) pair at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`FETCh:EDIagram:MEASview2:FDEViation?` might return #3160xxxx... (160-byte data) for the frequency deviation versus Time trace for measview2 of the Eye diagram.

FETCh:EDIagram[:MEASview<y>]:FERRor? (Query only)

Returns the frequency error in the eye diagram measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Eye diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Eye diagram).

Group

Fetch commands

Syntax

`FETCh:EDIagram[:MEASview<y>]:FERRor?`

Arguments

None

Returns

`<freq_error>::<NRf>` is the frequency error in Hz.

Examples

`FETCh:EDIagram:MEASview2:FERRor?` might return -10.7E+3, indicating the frequency error is -10.7 kHz for measview2 of the Eye diagram.

FETCh:EDIagram[:MEASview<y>]:I? (Query only)

Returns the I versus Time trace data in the eye diagram measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Eye diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Eye diagram).

Group

Fetch commands

Syntax

`FETCh:EDIagram[:MEASview<y>]:I?`

Arguments

None

Returns

`#<num_digit><num_byte><Y(1)><X(1)><Y(2)><X(2)>...<Y(n)><X(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<Y(n)><X(n)>` is the I level (normalized) and time (symbols) pair at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`FETCh:EDIagram:MEASview2:I?` might return #3160xxxx... (160-byte data) for the I versus Time trace for measview2 of the Eye diagram.

FETCh:EDIagram[:MEASview<y>]:Q? (Query only)

Returns the Q versus Time trace data in the eye diagram measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Eye diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Eye diagram).

Group

Fetch commands

Syntax

`FETCh:EDIagram[:MEASview<y>]:Q?`

Arguments

None

Returns

`#<num_digit><num_byte><Y(1)><X(1)><Y(2)><X(2)>...<Y(n)><X(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<Y(n)><X(n)>` is the Q level (normalized) and time (symbols) pair at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`FETCh:EDIagram:MEASview2:Q?` might return #3160xxxx... (160-byte data) for the Q versus Time trace for measview2 of the Eye digram.

FETCh:EMC:COUNt? (Query only)

Queries the total numbers of spots detected in EMC View.

Conditions

Measurement view: EMC - EMI Display

Group

Fetch commands

Syntax

`FETCh:EMC:COUNt?`

Arguments

None

Returns

<NR1>

Examples

`FETCH:EMC:COUNT?` might return `15`, indicating that 15 spots are detected.

FETCh:EMC:EMISsion:RESults:STATus? (Query Only)

Queries the status of the emission results in EMC view.

Conditions

Measurement view: EMC - EMI Display

Group

Fetch commands

Syntax

`FETCh:EMC:EMISsion:RESults:STATus?`

Arguments

None

Returns

<String> Either PASS or FAIL

Examples

`FETCH:EMC:EMISSION:RESULTS:STATUS?` returns `PASS`, indicating that the status of emission results is passed.

FETCh:EMC:SPOT<x>:AMBient:AMPLitude? (Query Only)

Queries the amplitude of the selected spot in ambient trace of EMC view.

<x> = the spot number

Conditions

Measurement view: EMC - EMI Display

Group

Fetch commands

Syntax

`FETCh:EMC:SPOT<x>:AMBIent:AMPLitude?`

Arguments

None

Returns

<NRf>

Examples

`FETCh:EMC:SPOT1:AMBIent:AMPLitude?` might return `52.612`, indicating that the amplitude of spot 1 in the ambient trace is 52.61 dBuV.

FETCh:EMC:SPOT<x>:AMBIent:DELTa? (Query Only)

Queries the Delta value of the selected spot in ambient trace of EMC View.

<x> = the spot number

Conditions

Measurement view: EMC - EMI Display

Group

Fetch commands

Syntax

`FETCh:EMC:SPOT<x>:AMBIent:DELTa?`

Arguments

None

Returns

<NRf>

Examples

`FETCh:EMC:SPOT2:AMBIent:DELTa?` might return `3.3614`, indicating that the delta value of spot 2 in the ambient trace is 3.36 dBuV.

FETCh:EMC:SPOT<x>:AMBIent:PASS? (Query Only)

Queries the status of the selected spot in ambient trace in EMC view. The default status is THRE.

<x> = the spot number

Conditions

Measurement view: EMC - EMI Display

Group

Fetch commands

Syntax

`FETCh:EMC:SPOT<x>:AMBIent:PASS?`

Arguments

None

Returns

`<String>` Either OLIM, OMAR or THRE

Where

`OLIM` = Over Limit

`OMAR` = Over Margin

`THRE` = Threshold

Examples

`FETCh:EMC:SPOT3:AMBIent:PASS?` might return `THRE`, indicating that spot 3 of the ambient trace is failing by crossing the threshold.

FETCh:EMC:SPOT<x>:FREQuency? (Query Only)

Queries the frequency of the specified spot in EMC View.

<x> = <NR1> indicates the spot number

Conditions

Measurement view: EMC - EMI Display

Group

Fetch commands

Syntax

`FETCh:EMC:SPOT<x>:FREQuency?`

Arguments

None

Returns

<NRf>

Examples

`FETCh:EMC:SPOT2:FREQuency?` might return `366.12500000E+3`, indicating that the frequency of spot 2 is 366.125 kHz.

FETCh:EMC:SPOT<x>:PASS? (Query Only)

Queries the overall status of the specified spot in EMC view. The default status is THRE.

<x> = <NR1> indicates the spot number.

Conditions

Measurement view: EMC - EMI Display

Group

Fetch commands

Syntax

`FETCh:EMC:SPOT<x>:PASS?`

Arguments

None

Returns

`<String>` Either OLIM, OMAR or THRE

where

`OLIM` = Over Limit

`OMAR` = Over Margin

`THRE` = Threshold

Examples

`FETCh:EMC:SPOT1:PASS?` might return `OLIM`, indicating that spot 1 is failing by crossing over the limit.

FETCh:EMC:SPOT<x>:RANGe? (Query Only)

Queries the range of the specified spot in EMC View.

<x> = <NR1> indicates the spot number.

Conditions

Measurement view: EMC - EMI Display

Group

Fetch commands

Syntax

`FETCh:EMC:SPOT<x>:RANGe?`

Arguments

None

Returns

<String> from A to T

Examples

`FETCh:EMC:SPOT1:RANGe?` might return `"A"`, indicating that the range of spot 1 is "A".

FETCh:EMC:SPOT<x>:RECAll:AMPLitude? (Query Only)

Queries the amplitude of the selected spot in recalled trace of EMC View.

<x> = the spot number

Conditions

Measurement view: EMC - EMI Display

Group

Fetch commands

Syntax

`FETCh:EMC:SPOT<x>:RECAll:AMPLitude?`

Arguments

None

Returns

<NRf>

Examples

`FETCh:EMC:SPOT1:RECAll:AMPLitude?` might return 23.598, indicating that the amplitude of spot 1 in the recalled trace is 23.59 dBuV.

FETCh:EMC:SPOT<x>:RECAll:DELTa? (Query Only)

Queries the Delta value of the specified spot in recalled trace of EMC View.

<x> = the spot number

Conditions

Measurement view: EMC - EMI Display

Group

Fetch commands

Syntax

`FETCh:EMC:SPOT<x>:RECAll:DELTa?`

Arguments

None

Returns

<NRf>

Examples

`FETCh:EMC:SPOT2:RECAll:DELTa?` might return 5.5878, indicating that delta value of the spot 2 in the recalled trace is 5.58 dBuV.

FETCh:EMC:SPOT<x>:RECAll:PASS? (Query Only)

Queries the status of the selected spot in recalled trace in EMC view. The default status is THRE.

<x> = the spot number

Conditions

Measurement view: EMC - EMI Display

Group

Fetch commands

Syntax

`FETCh:EMC:SPOT<x>:RECAll:PASS?`

Arguments

None

Returns

`<String>` Either OLIM, OMAR or THRE

Where

`OLIM` = Over Limit

`OMAR` = Over Margin

`THRE` = Threshold

Examples

`FETCh:EMC:SPOT1:RECAll:PASS?` might return `OMAR`, indicating that spot 1 of the recalled trace is failing by crossing the Margin.

FETCh:EMC:SPOT<x>:SCDEtector<y>:AMPLitude? (Query Only)

Queries the amplitude of the specified spot with respect to specified scan detector in the results table.

<x> = the spot number

<y> = the scan detector number (either 1 or 2)

Conditions

Measurement view: EMC - EMI Display

Group

Fetch commands

Syntax

`FETCh:EMC:SPOT<x>:SCDEtector<y>:AMPLitude?`

Arguments

None

Returns

<NRf>

Examples

`FETCh:EMC:SPOT1:SCDEtector1:AMPLitude?` might return `20.6209`, indicating that the amplitude of spot 1 for scan detection 1 is 20.62 dBuV.

FETCh:EMC:SPOT<x>:SCDEtector<y>:DELTa? (Query Only)

Queries the Delta value of the specified spot with respect to specified scan detector in the results table.

<x> = the spot number

<y> = the scan detector number (either 1 or 2)

Conditions

Measurement view: EMC - EMI Display

Group

Fetch commands

Syntax

`FETCh:EMC:SPOT<x>:SCDEtector<y>:DELTa?`

Arguments

None

Returns

<NRf>

Examples

`FETCh:EMC:SPOT2:SCDEtector1:DELTa?` might return `3.6209`, indicating that the delta value of spot 2 for scan detection 1 is 3.62 dBuV.

FETCh:EMC:SPOT<x>:SCDEtector<y>:PASS? (Query Only)

Queries the status of the specified spot based on the specified scan detector in EMC view. The default status is THRE.

<x> = the spot number

<y> = the scan detector number (either 1 or 2)

Conditions

Measurement view: EMC - EMI Display

Group

Fetch commands

Syntax

`FETCh:EMC:SPOT<x>:SCDEtector<y>:PASS?`

Arguments

None

Returns

`<String>` either OLIM, OMAR or THRE

where

`OLIM` = Over Limit

`OMAR` = Over Margin

`THRE` = Threshold

Examples

`FETCh:EMC:SPOT2:SCDEtector1:PASS?` might return `THRE`, indicating that spot 2 of scan detector 1 is failing by crossing the threshold.

FETCh:EMC:SPOT<x>:SPDEtector<y>:AMPLitude? (Query Only)

Queries the amplitude of the specified spot with respect to specified spot detector in the results table.

<x> = the spot number

<y> = the spot detector number (either 1, 2, or 3)

Conditions

Measurement view: EMC - EMI Display

Group

Fetch commands

Syntax

`FETCh:EMC:SPOT<x>:SPDEtector<y>:AMPLitude?`

Arguments

None

Returns

<NRf>

Examples

`FETCh:EMC:SPOT1:SPDEtector1:AMPLitude?` might return `18.365`, indicating that amplitude of spot 1 for spot detection 1 is 18.36 dBuV.

FETCh:EMC:SPOT<x>:SPDEtector<y>:DELTa? (Query Only)

Queries the Delta value of the specified spot with respect to specified spot detector in the results table.

<x> = the spot number

<y> = the spot detector number (either 1, 2, or 3)

Conditions

Measurement view: EMC - EMI Display

Group

Fetch commands

Syntax

`FETCh:EMC:SPOT<x>:SPDEtector<y>:DELTa?`

Arguments

None

Returns

<NRf>

Examples

`FETCh:EMC:SPOT2:SPDEtector1:DELTa?` might return `2.2578`, indicating that delta value of spot 2 for spot detection 1 is 2.25 dBuV.

FETCh:EMC:SPOT<x>:SPDEtector<y>:PASS? (Query Only)

Queries the status of the selected spot based on the specified spot detector in EMC view. The default status is THRE.

`<x>` = the spot number

`<y>` = the spot detector number (either 1, 2, or 3)

Conditions

Measurement view: EMC - EMI Display

Group

Fetch commands

Syntax

`FETCh:EMC:SPOT<x>:SPDEtector<y>:PASS?`

Arguments

None

Returns

<String> Either OLIM, OMAR or THRE

where

`OLIM` = Over Limit

`OMAR` = Over Margin

`THRE` = Threshold

Examples

`FETCh:EMC:SPOT1:SPDEtector2:PASS?` might return OMAR, indicating that the spot 1 of spot detector 2 is failing by crossing Margin.

FETCh:EVM[:MEASview<y>]:FERRor? (Query only)

Returns the frequency error in the `EVM` versus Time measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: EVM versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod EVM versus Time).

Group

Fetch commands

Syntax

`FETCh:EVM[:MEASVIEW<y>]:FERRor?`

Arguments

None

Returns

`<freq_error>::<NRf>` is the frequency error in Hz.

Examples

`FETCh:EVM:MEASview2:FERRor?` might return -10.7E+3, indicating the frequency error is -10.7 kHz for measview2 of EVM.

FETCh:EVM[:MEASview<y>]:PEAK? (Query only)

Returns the peak value in the **EVM** versus Time measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: EVM versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod EVM versus Time).

Group

Fetch commands

Syntax

`FETCh:EVM[:MEASview<y>]:PEAK?`

Related Commands

FETCh:EVM[:MEASVIEW<y>]:PINDex? (on page 1155)

Arguments

None

Returns

`<peak>::<NRf>` is the peak EVM value in percent (%).

Examples

`FETCh:EVM:MEASview2:PEAK?` might return 1.32, indicating the peak EVM value is 1.32% for measview2 of EVM.

FETCh:EVM[:MEASview<y>]:PINDex? (Query only)

Returns the time at the **EVM** peak.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: EVM versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod EVM versus Time).

Group

Fetch commands

Syntax

`FETCh:EVM[:MEASview<y>]:PINDex?`

Related Commands

FETCh:EVM[:MEASview<y>]:PEAK? (on page 1155)

Arguments

None

Returns

`<peak_time>::<NRf>` is the time at the EVM peak in symbol number.

The unit can be changed by the `[SENSe]:DDEMod:TIMe:UNITs` command.

Examples

`FETCh:EVM:MEASview2:PINDex` might return 68.000, indicating that the EVM peak is at symbol #68 for measview2 of EVM.

FETCh:EVM[:MEASview<y>]:RMS? (Query only)

Returns the RMS (Root-Mean-Square) value in the **EVM** versus Time measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: EVM versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod EVM versus Time).

Group

Fetch commands

Syntax

`FETCh:EVM[:MEASview<y>]:RMS?`

Arguments

None

Returns

`<rms>::<NRf>` is the RMS EVM value in percent (%).

Examples

`FETCh:EVM:MEASview2:RMS?` might return 0.582, indicating the RMS EVM value is 0.582% for the measview2 of EVM.

FETCh:EVM[:MEASview<y>]:TRACe? (Query only)

Returns the **EVM** versus Time trace data.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: EVM versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod EVM versus Time).

Group

Fetch commands

Syntax

`FETCh:EVM[:MEASview<y>]:TRACe?`

Arguments

None

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the EVM value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`FETCh:EVM:MEASview2:TRACe?` might return #42036xxxx... (2036-byte data) for the EVM versus Time Measview 2.

FETCh:FDVTime[:MEASview<y>]:FERRor? (Query only)

Returns the frequency error in the Frequency deviation versus Time measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Frequency deviation versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Frequency deviation versus Time).

Group

Fetch commands

Syntax

`FETCh:FDVTime:MEASview<y>:FERRor?`

Arguments

None

Returns

<freq_error>::<NRf> is the frequency error in Hz.

Examples

FETCH:FDVTime:MEASview2:FERROR? might return -10.7E+3, indicating the frequency error is -10.7 kHz.

FETCh:FDVTime[:MEASview<y>]:TRACe? (Query only)

Returns the Frequency deviation versus Time trace data.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Frequency deviation versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Frequency deviation versus Time).

Group

Fetch commands

Syntax

FETCh:FDVTime:MEASview<y>:TRACe?

Arguments

None

Returns

#<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the frequency deviation value at the n^{th} data point in percent (Hz), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

FETCH:FDVTime:MEASview2:TRACE? might return #3160xxxx... (160-byte data) for the Frequency deviation versus Time Measview 2.

FETCh:{FM|PM}:FERRor? (Query Only)

Returns the frequency error in the Frequency modulation or Phase modulation measurements.

Conditions

Measurement views: Frequency deviation versus Time

Group

Fetch commands

Syntax
`FETCh:{FM|PM}:FERRor?`

Arguments
None

Returns
`<freq_error>::<NRf>` is the frequency error in Hz.

Examples
`FETCH:{FM|PM}:FERROR?` might return -10.7E+3, indicating the frequency error is -10.7 kHz.

FETCh:FM:PHALf? (Query Only)
Returns the half peak-peak frequency deviation (Pk-Pk/2) in the FM measurement.

Conditions
Measurement views: FM

Group
Fetch commands

Syntax
`FETCh:FM:PHALf?`

Arguments
None

Returns
`<Pk-Pk/2>::<NRf>` is the half peak-peak frequency deviation in Hz.

Examples
`FETCH:FM:PHALF?` might return 628.9E+3, indicating the half peak-peak frequency deviation is 628.9 kHz.

FETCh:FM:PNEGative? (Query Only)
Returns the negative peak frequency deviation (–Pk) in the FM measurement.

Conditions
Measurement views: FM

Group
Fetch commands

Syntax
`FETCh:FM:PNEGative?`

Arguments
None

Returns

<-Pk>::<NRf> is the negative peak frequency deviation in Hz.

Examples

FETCH:FM:PNEGATIVE? might return –495.6E+3, indicating the negative peak frequency deviation is –495.6 kHz.

FETCh:FM:PPOSitive? (Query Only)

Returns the positive peak frequency deviation (+Pk) in the FM measurement.

Conditions

Measurement views: FM

Group

Fetch commands

Syntax

FETCh:FM:PPOSitive?

Arguments

None

Returns

<+Pk>::<NRf> is the positive peak frequency deviation in Hz.

Examples

FETCH:FM:PPOSITIVE? might return 763.2E+3, indicating the positive peak frequency deviation is 763.2 kHz.

FETCh:FM:PTPeak? (Query Only)

Returns the peak-peak frequency deviation (Pk-Pk) in the FM measurement.

Conditions

Measurement views: FM

Group

Fetch commands

Syntax

FETCh:FM:PTPeak?

Arguments

None

Returns

<Pk-Pk>::<NRf> is the peak-peak frequency deviation in Hz.

Examples

FETCH:FM:PTPEAK? might return 1.258E+6, indicating the peak-peak frequency deviation is 1.258 MHz.

FETCh:FM:RESult? (Query Only)

Returns the FM measurement results.

Conditions

Measurement views: FM

Group

Fetch commands

Syntax

`FETCh:FM:RESult?`

Arguments

None

Returns

`<+Pk>,<-Pk>,<RMS>,<Pk-Pk>,<Pk-Pk/2>`

Where

`<+Pk>::<NRf>` is the positive peak frequency deviation in Hz.

`<-Pk>::<NRf>` is the negative peak frequency deviation in Hz.

`<RMS>::<NRf>` is the RMS frequency deviation in Hz.

`<Pk-Pk>::<NRf>` is the peak-peak frequency deviation in Hz.

`<Pk-Pk/2>::<NRf>` is the half peak-peak frequency deviation in Hz.

Examples

`FETCH:FM:RESULT?` might return `763.2E+3,-494.6E+3,271.2E+3,1.258E+6,628.9E+3`.

FETCh:FM:RMS? (Query Only)

Returns the RMS frequency deviation in the FM measurement.

Conditions

Measurement views: FM

Group

Fetch commands

Syntax

`FETCh:FM:RMS?`

Arguments

None

Returns

`<RMS>::<NRf>` is the RMS frequency deviation in Hz.

Examples

FETCH:FM:RMS? might return 271.2E+3, indicating the RMS frequency deviation is 271.2 kHz.

FETCh:{FSETtling|PSETtling}:FTTime? (Query Only)

Returns the settling time from the trigger position in seconds. The settling time is measured from the trigger point (see :FETCh:FSETtling:TRIGger:TIME).

Conditions

Measurement views: Frequency and Phase Settling

Group

Fetch commands

Syntax

FETCh:{FSETtling|PSETtling}:FTTime?

Arguments

None

Returns

<value>::<NRf> is the settling time in seconds.

Examples

FETCH:FSETTLING:FTTIME? might return 44.8300E-6, indicating the settling time is 44.83 µs.

FETCh:{FSETtling|PSETtling}:MASK[:PASS]? (Query Only)

Returns whether the input signal passes the mask test.

Conditions

Measurement views: Frequency and Phase Settling

Group

Fetch commands

Syntax

FETCh:{FSETtling|PSETtling}:MASK[:PASS]?

Arguments

None

Returns

0 is returned if the signal fails the mask test.

1 is returned if the signal passes the mask test.

Examples

FETCH:FSETTLING:MASK:PASS? might return 1, indicating the signal passed (did not exceed the mask limits).

FETCh:{FSETtling|PSETtling}:SETTled:FREQuency? (Query Only)

Returns the frequency at which the signal is considered settled.

Conditions

Measurement views: Frequency and Phase Settling

Group

Fetch commands

Syntax

`FETCh:{FSETtling|PSETtling}:SETTled:FREQuency?`

Arguments

None

Returns

`<value>::<NRf>` is the frequency at which the signal is settled.

Examples

`FETCH:FSETTLING:SETTLED:FREQUENCY?` might return `2.44838E+9`, indicating the input signal frequency at the point where the signal is considered settled is 2.44838 GHz.

FETCh:{FSETtling|PSETtling}:SETTled[:PASS]? (Query Only)

Returns whether the input signal is settled with the tolerance range.

Conditions

Measurement views: Frequency and Phase Settling

Group

Fetch commands

Syntax

`FETCh:{FSETtling|PSETtling}:SETTled[:PASS]?`

Arguments

None

Returns

`0` is returned if the signal is not settled within the tolerance range.

`1` is returned if the signal is settled within the tolerance range.

Examples

`FETCH:FSETTLING:SETTLED:PASS?` might return `1`, indicating the signal is settled within the tolerance range.

FETCh:{FSETtling|PSETtling}:SETTled:TIME? (Query Only)

Returns the settled time in seconds. The settled time is measured from the measurement start point.

Conditions

Measurement views: Frequency and Phase Settling

Group

Fetch commands

Syntax

`FETCh:{FSETtling|PSETtling}:SETTled:TIME?`

Arguments

None

Returns

`<value>::<NRf>` is the settled time in seconds.

Examples

FETCH:FSETTLING:SETTLED:TIME? might return `299.830000E-6`, indicating the settled time is 299.83 µs.

FETCh:{FSETtling|PSETtling}:SLMSd[:PASS]? (Query Only)

Returns whether the input signal is settled within the specified tolerance and the signal is settled longer than the Minimum Settled Duration (`[SENSe]:{FSETtling|PSETtling}:SDURation:MINimum`).

Conditions

Measurement views: Frequency and Phase Settling

Group

Fetch commands

Syntax

`FETCh:{FSETtling|PSETtling}:SLMSd[:PASS]?`

Arguments

None

Returns

`0` is returned if the signal is not settled within the tolerance range and minimum settled duration.

`1` is returned if the signal is settled within the tolerance range and minimum settled duration.

Examples

`FETCH:FSETTLING:SLMSD:PASS?` might return 1, indicating the signal is settled within the tolerance range and the minimum settled duration.

FETCh:{FSETtling|PSETtling}:STARt:TIME? (Query Only)

Returns the start time for the measurement in seconds. The start time is measured from the start of the analysis period.

Conditions

Measurement views: Frequency and Phase Settling

Group

Fetch commands

Syntax

```
FETCh:{FSETtling|PSETtling}:STARt:TIME?
```

Arguments

None

Returns

<value>::<NRf> is the time in seconds when the measurement started.

Examples

FETCH:FSETTLING:START:TIME? might return 251.4300E-6, indicating the measurement started 251.43 µs after the beginning of the analysis period.

FETCh:{FSETtling|PSETtling}:TIME? (Query Only)

Returns the settling time in seconds. The settling time is measured from the start time (see :FETCh:FSETtling:STARt:TIME).

Conditions

Measurement views: Frequency and Phase Settling

Group

Fetch commands

Syntax

```
FETCh:{FSETtling|PSETtling}:TIME?
```

Arguments

None

Returns

<value>::<NRf> is the settling time in seconds.

Examples

FETCH:FSETTLING:TIME? might return 48.4000E-6, indicating the settling time is 48.4 µs.

FETCh:{FSETtling|PSETtling}:TRACe<x>:X? (Query Only)

Returns the Frequency or Phase values of the specified trace.

The parameter <x> = 1 and 2, representing Trace 1 and Trace 2, respectively.

Conditions

Measurement views: Frequency and Phase Settling

Group

Fetch commands

Syntax

`FETCh:{FSETtling|PSETtling}:TRACe<x>:X?`

Arguments

None

Returns

`#<num_digit><num_byte><y(1)><y(2)>...<y(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<y(n)>` is the frequency (Hz) or phase (degrees) at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`FETCH:FSETTLING:TRACE1:X?` might return `#574232xxxx...` (74232-byte data) for the frequency values of Trace 1.

FETCh:{FSETtling|PSETtling}:TRACe<x>:XY? (Query Only)

Returns the time and frequency or phase value pairs of the specified trace. The parameter <x> = 1 and 2, representing Trace 1 and Trace 2, respectively.

Conditions

Measurement views: Frequency and Phase Settling

Group

Fetch commands

Syntax

`FETCh:{FSETtling|PSETtling}:TRACe<x>:XY?`

Arguments

None

Returns

`#<num_digit><num_byte><x(1)><y(1)><x(2)><y(2)>...<x(n)><y(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<x(n)><y(n)>` is the time and frequency or phase value pair at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`FETCH:FSETTLING:TRACE2:XY?` might return `#574232xxxx...` (74232-byte data) for the time and frequency pairs of the Trace 2.

FETCh:{FSETtling|PSETtling}:TRACe<x>[:Y]? (Query Only)

Returns the frequency or phase values of the specified trace.

The parameter <x> = 1 and 2, representing Trace 1 and Trace 2, respectively.

Conditions

Measurement views: Frequency and Phase Settling

Group

Fetch commands

Syntax

```
FETCh:{FSETtling|PSETtling}:TRACe<x>[:Y]?
```

Arguments

None

Returns

`#<num_digit><num_byte><y(1)><y(2)>...<y(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<y(n)>` is the frequency or phase value at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`FETCH:FSETTLING:TRACE1:Y?` might return `#575148xxxx...` (75148-byte data) for the frequency values of Trace 1.

FETCh:{FSETtling|PSETtling}:TRIGger:TIME? (Query Only)

Returns the time when the trigger occurred in seconds.

Conditions

Measurement views: Frequency and Phase Settling

Group

Fetch commands

Syntax

```
FETCh:{FSETtling|PSETtling}:TRIGger:TIME?
```

Arguments

None

Returns

`<value>::<NRf>` is the time in seconds after the measurement start point when the trigger occurred.

Examples

`FETCH:FSETTLING:TRIGGER:TIME?` might return `2.255E-6`, indicating the time when the trigger occurred 255 µs after the start of the measurement.

FETCh:{FSETtling|PSETtling}:VALue? (Query Only)

Returns the settled value in Hz for Frequency Settling and in degrees for Phase Settling.

Conditions

Measurement views: Frequency and Phase Settling

Group

Fetch commands

Syntax

`FETCh:{FSETtling|PSETtling}:VALue?`

Arguments

None

Returns

`<value>::<NRf>` is the settling value in Hz for Frequency Settling and in degrees for Phase Settling.

Examples

`FETCH:FSETTLING:VALUE?` might return `2.44838155E+9`, indicating the settled frequency is 2.44838 GHz.

FETCh:FSETtling:ERRor? (Query Only)

Returns the settled error in Hz in Frequency Settling. Only for Frequency Settling Time measurement. In Frequency Settling:

- When Target Reference is set to Auto, Settled Error = 0.

Conditions

Measurement views: Frequency Settling

Group

Fetch commands

Syntax

`FETCh:FSETtling:ERRor?`

Arguments

None

Returns

`<value>::<NRf>` returns the settled error in Hz.

When Target Reference is set to Auto, Settled Error = 0.

When Target Reference is set to Meas Freq:

Settled Error = Settled Frequency – (Measurement Frequency + Offset)

Examples

`FETCH:FSETTLING:FTTIME?` might return `0`, indicating the Target Reference is set to Auto.

FETCh:FVTime[:MEASview<y>]? (Query Only)

Returns the Frequency versus Time trace data.

Conditions

Measurement views: Frequency versus Time

Group

Fetch commands

Syntax

`FETCh:FVTime[:MEASview<y>]?`

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the frequency in Hz at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`FETCH:FVTIME:MEASview1?` might return `#3156xxxx...` (156-byte data) for the Frequency versus Time trace.

FETCh:FVTime[:MEASview<y>]:MAXimum? (Query Only)

Returns the maximum value in the Frequency versus Time measurement.

Conditions

Measurement views: Frequency versus Time

Group

Fetch commands

Syntax

`FETCh:FVTime[:MEASview<y>]:MAXimum?`

Related Commands

FETCh:FVTime[:MEASview<y>]:MAXLocation? (on page 1170)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

<max>::<NRf> is the maximum frequency drift in Hz.

Examples

FETCH:FVTIME:MEASview1:MAXIMUM? might return 2.625E+6, indicating the maximum frequency drift is 2.625 MHz.

FETCh:FVTime[:MEASview<y>]:MAXLocation? (Query Only)

Returns the time at which the frequency drift is maximum.

Conditions

Measurement views: Frequency versus Time

Group

Fetch commands

Syntax

FETCh:FVTime[:MEASview<y>]:MAXLocation?

Related Commands

FETCh:FVTime[:MEASview<y>]:MAXimum? (on page 1169)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

<max_time>::<NRf> is the time in seconds at which the frequency drift is maximum.

Examples

FETCH:FVTIME:MEASview1:MAXLOCATION? might return 25.03E-9, indicating the frequency drift is maximum at 25.03 ns.

FETCh:FVTime[:MEASview<y>]:MINimum? (Query Only)

Returns the minimum value in the Frequency versus Time measurement.

Conditions

Measurement views: Frequency versus Time

Group

Fetch commands

Syntax

FETCh:FVTime[:MEASview<y>]:MINimum?

Related Commands

FETCh:FVTime[:MEASview<y>]:MINLocation? (on page 1171)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

<min>::<NRf> is the minimum frequency drift in Hz.

Examples

FETCH:FVTIME:MEASview1:MINIMUM? might return -6.618E+6, indicating the minimum frequency drift is -6.618 MHz.

FETCh:FVTime[:MEASview<y>]:MINLocation? (Query Only)

Returns the time at which the frequency drift is minimum.

Conditions

Measurement views: Frequency versus Time

Group

Fetch commands

Syntax

FETCh:FVTime[:MEASview<y>]:MINLocation?

Related Commands

FETCh:FVTime[:MEASview<y>]:MINimum? (on page 1170)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

<min_time>::<NRf> is the time in seconds at which the frequency drift is minimum.

Examples

FETCH:FVTIME:MEASview1:MINLOCATION? might return 450.7E-9, indicating the frequency drift is minimum at 450.7 ns.

FETCh:FVTime[:MEASview<y>]:OGRAm? (Query only)

Returns trace data of a line in the ogram. The line is selected using the TRACe:FVTime[:MEASview<y>]:OGRAm:SELect:LINE command.

Conditions

Applicable measurement view must be active.

Group

Fetch commands

Syntax

`FETCh:FVTime[:MEASview<y>]:OGRAm?`

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional. By default y is 1.

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in <num_byte>.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the frequency at the n^{th} data point in Hz, represented as a 4-byte little-endian IEEE floating-point value.

Examples

`FETCh:FVTime:MEASview1:OGRAm?` might return #43204xxxx... (3204-byte data) for the line in the ogram.

FETCh:FVTime[:MEASview<y>]:RESult? (Query Only)

Returns the Frequency versus Time measurement results.

Conditions

Measurement views: Frequency versus Time

Group

Fetch commands

Syntax

`FETCh:FVTime[:MEASview<y>]:RESult?`

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

<max>,<max_time>,<min>,<min_time>

Where

`<max>::<NRf>` is the maximum frequency drift in Hz.

`<max_time>::<NRf>` is the time in seconds at which the frequency drift is maximum.

`<min>::<NRf>` is the minimum frequency drift in Hz.

`<min_time>::<NRf>` is the time in seconds at which the frequency drift is minimum.

Examples

`FETCH:FVTIME:MEASview1:RESULT?` might return `2.625E+6,25.03E-9,-6.618E+6,450.7E-9`, indicating the maximum frequency drift is 2.625 MHz at 25.03 ns and the minimum frequency drift is -6.618 MHz at 450.7 ns.

FETCh:FVTime[:MEASview<y>]:TRACe:BITMap? (Query Only)

Returns trace waveform data of the bitmap trace in the Frequency versus Time measurement.

Conditions

Measurement views: Frequency versus Time

Group

Fetch commands

Syntax

`FETCh:FVTime[:MEASview<y>]:TRACe:BITMap?`

Arguments

None

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Returns

A binary block of 32-bit floating point values (IEEE standard single precision floating point format). Each value ranges from 0 to 1.0, where 1.0 represents 100%.

Bitmaps have 201 rows. Data[0] is the first column in the first row, and the following data continues along that row before moving to the next.

Bitmap values are returned in the order of top left pixel first, then the rest of the top row, after that row by row.
In other words, the pixels in the bitmap are returned from left to right and top to bottom; one row at a time.

Examples

`FETCH:FVTIME:TRACE:BITMAP?` might return #6644004xxxx...

FETCh:IQVTime[:MEASview<y>]:I? (Query Only)

Returns the I versus Time trace data.

Conditions

Measurement views: RF I&Q versus Time

Group

Fetch commands

Syntax

`FETCh:IQVTime[:MEASview<y>]:I?`

Arguments

None

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the I level in volts at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

FETCH:IQVTIME:MEASview1:I? might return #3160xxxx... (160-byte data) for the I versus Time trace.

FETCh:IQVTime[:MEASview<y>]:MAXimum (Query Only)

Returns the maximum value in the RF I&Q versus Time measurement.

Conditions

Measurement views: RF I&Q versus Time

Group

Fetch commands

Syntax

FETCh:IQVTime[:MEASview<y>]:MAXimum

Related Commands

FETCh:IQVTime[:MEASview<y>]:MAXLocation (on page 1174)

Arguments

None

Returns

<max>::<NRf> is the maximum I or Q level in volts.

Use the TRACe:IQVTime[:MEASview<y>]:SELect:I or TRACe:IQVTime[:MEASview<y>]:SELect:Q command to select the trace.

Examples

FETCh:IQVTime:MEASview1:MAXimum? might return 1.214, indicating the maximum I or Q level is 1.214 V.

FETCh:IQVTime[:MEASview<y>]:MAXLocation (Query Only)

Returns the time at which the I or Q level is maximum.

Conditions

Measurement views: RF I&Q versus Time

Group

Fetch commands

Syntax

FETCh:IQVTime[:MEASview<y>]:MAXLocation

Related Commands

FETCh:IQVTime[:MEASview<y>]:MAXimum (on page 1174)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

<max_time>::<NRf> is the time in seconds at which the I or Q level is maximum.

Examples

FETCh:IQVTime:MEASview1:MAXLocation might return 175.3E-9, indicating the I or Q level is maximum at 175.3 ns.

FETCh:IQVTime[:MEASview<y>]:MINimum (Query Only)

Returns the minimum value in the RF I&Q versus Time measurement.

Conditions

Measurement views: RF I&Q versus Time

Group

Fetch commands

Syntax

FETCh:IQVTime[:MEASview<y>]:MINimum

Related Commands

FETCh:IQVTime[:MEASview<y>]:MINLocation (on page 1175)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

<min>::<NRf> is the minimum I or Q level in volts.

Use the TRACe:IQVTime[:MEASview<y>]:SELect:I or TRACe:IQVTime[:MEASview<y>]:SELect:Q command to select the trace.

Examples

FETCh:IQVTime:MEASview1:MINimum? might return -370.5E-3, indicating the minimum I or Q level is -370.5 mV.

FETCh:IQVTime[:MEASview<y>]:MINLocation (Query Only)

Returns the time at which the I or Q level is minimum.

Conditions

Measurement views: RF I&Q versus Time

Group

Fetch commands

Syntax

`FETCh:IQVTime[:MEASview<y>]:MINLocation`

Related Commands

FETCh:IQVTime[:MEASview<y>]:MINimum (on page 1175)

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

`<min_time>::<NRf>` is the time in seconds at which the I or Q level is minimum.

Examples

`FETCh:IQVTime:MEASview1:MINLocation` might return `450.7E-9`, indicating the I or Q level is minimum at 450.7 ns.

FETCh:IQVTime[:MEASview<y>]:Q (Query Only)

Returns the Q versus Time trace data.

Conditions

Measurement views: RF I&Q versus Time

Group

Fetch commands

Syntax

`FETCh:IQVTime[:MEASview<y>]:Q`

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the Q level in volts at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`FETCh:IQVTime:MEASview1:Q?` might return `#3160xxxx...` (160-byte data) for the Q versus Time trace.

FETCh:IQVTime[:MEASview<y>]:RESult (Query Only)

Returns the RF I&Q versus Time measurement results.

Conditions

Measurement views: RF I&Q versus Time

Group

Fetch commands

Syntax

`FETCh:IQVTime[:MEASview<y>]:RESult`

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

`<max>,<max_time>,<min>,<min_time>`

Where

`<max>::<NRf>` is the maximum I or Q level in volts.

`<max_time>::<NRf>` is the time in seconds at which the I or Q level is maximum.

`<min>::<NRf>` is the minimum I or Q level in volts.

`<min_time>::<NRf>` is the time in seconds at which the I or Q level is minimum.

Use the `TRACe:IQVTime[:MEASview<y>]:SELect:I` or `TRACe:IQVTime:MEASview<y>:SELect:Q` command to select the trace.

Examples

`FETCh:IQVTime:MEASview1:RESult?` might return `1.214,175.3E-9,-370.5E-3,450.7E-9`, indicating the maximum I or Q level is 1.214 V at 175.3 ns and the minimum I or Q level is -370.5 mV at 450.7 ns.

FETCh:LTE:ACLR:MHITs:COUNt? (Query Only)

Returns the number of rows in the results table of the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

Group

Fetch commands

Syntax

`FETCh:LTE:ACLR:MHITs:COUNt?`

Returns

<NR1>

Examples

`FETCH:LTE:ACLR:MHITS:COUNT?` might return 7, indicating there are seven rows in the results table.

FETCh:LTE:ACLR:MHITs<x>:BANDwidth? (Query Only)

Returns the bandwidth value of the channel (row) of the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

The parameter <x> is the row number in the LTE ACLR display results table.

Group

Fetch commands

Syntax

`FETCh:LTE:ACLR:MHITs<x>:BANDwidth?`

Returns

<NR3>

Examples

`FETCh:LTE:ACLR:MHITS1:BANDWIDTH?` might return 4.5150000000E+6, which indicates a bandwidth for Channel 1 of 4.515 MHz.

FETCh:LTE:ACLR:MHITs<x>:CHANnel:NAME? (Query Only)

Returns the channel name of the specified index in the results table of the LTE ACLR display. The minimum index will be 1 and the maximum index will be the count of the channels in the results table.

Conditions

Measurement view: LTE ACLR

The parameter <x> is a positive integer and represents the row number in the LTE ACLR display results table.

Group

Fetch commands

Syntax

`FETCh:LTE:ACLR:MHITs<x>:CHANnel:NAME?`

Returns

<string>

Examples

`FETCH:LTE:ACLR:MHITS2:CHANNEL:NAME?` might return `"B"`, which is the name of channel 2.

FETCh:LTE:ACLR:MHITs<x>:FREQuency:OFFSet? (Query Only)

Returns the frequency offset of the specified channel (row) of the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

The parameter <x> is a positive integer and represents the row number in the LTE ACLR display results table.

Group

Fetch commands

Syntax

```
FETCh:LTE:ACLR:MHITs<x>:FREQuency:OFFSet?
```

Returns

<NR3>

Examples

`FETCh:LTE:ACLR:MHITS1:FREQUENCY:OFFSET?` might return `5.0000000000E+6` which indicates the frequency offset of Channel 1 is 5 MHz.

FETCh:LTE:ACLR:MHITs<x>:INTeg:LOWEr:ABSolute? (Query Only)

Returns the lower absolute value of the channel (row) of the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

The parameter <x> is the row number in the LTE ACLR display results table.

Group

Fetch commands

Syntax

```
FETCh:LTE:ACLR:MHITs<x>:INTeg:LOWEr:ABSolute?
```

Returns

<NRf>

Examples

`FETCh:LTE:ACLR:MHITS1:INTEG:LOWER:ABSOLUTE?` might return `-53.3921980303`, which indicates the lower absolute value of Channel 1 is -53.3921980303 dBm.

NOTE: The actual units depends on the choice made from the units tab of the Analysis control panel.

FETCh:LTE:ACLR:MHITs<x>:INTeg:LOWEr:RELative? (Query Only)

Returns the lower relative value of the channel (row) of the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

The parameter <x> is the row number in the LTE ACLR display results table.

Group

Fetch commands

Syntax

```
FETCh:LTE:ACLR:MHITs<x>:INTeg:LOWEr:RELative?
```

Returns

<NRf>

Examples

FETCh:LTE:ACLR:MHITS1:INTEG:LOWER:RELATIVE? might return –53.3921980303, which indicates lower relative value of Channel 1 is -53.3921980303 dB.

FETCh:LTE:ACLR:MHITs<x>:INTeg:UPPEr:ABSolute? (Query Only)

Returns the upper absolute value of the channel (row) of the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

The parameter <x> is a positive integer and represents the row number in the LTE ACLR display results table.

Group

Fetch commands

Syntax

FETCh:LTE:ACLR:MHITs<x>:INTeg:UPPEr:ABSolute?

Returns

<NRf>

Examples

FETCH:LTE:ACLR:MHITS1:INTEG:UPPER:ABSOLUTE? might return –53.3921980303, which indicates upper absolute value of Channel 1 is -53.3921980303 dBm.

FETCh:LTE:ACLR:MHITs<x>:INTeg:UPPEr:RELative? (Query Only)

Returns the upper relative value of the channel (row) of the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

The parameter <x> is the row number in the LTE ACLR display results table.

Group

Fetch commands

Syntax

FETCh:LTE:ACLR:MHITs<x>:INTeg:UPPEr:RELative?

Returns

<NRf>

Examples

READ:LTE:ACLR:MHITS1:INTEG:UPPER:RELATIVE? might return –53.3921980303, which indicates upper relative value of Channel 1 is -53.3921980303 dB.

FETCh:LTE:ACLR:REFerence:POWer? (Query Only)

Returns the reference power level in the LTE ACLR measurement.

Conditions

Measurement view: LTE ACLR

Group

Fetch commands

Syntax

`FETCh:LTE:ACLR:REFerence:POWer?`

Returns

<NRf>

Examples

`FETCH:LTE:ACLR:REFERENCE:POWER?` might return `-16.92`, indicating that the reference power level is –16.92 dBm.

FETCh:LTE:ACLR:RESUlts:STATus? (Query Only)

Returns the pass or fail status of the LTE ACLR display measurement.

Conditions

Measurement view: LTE ACLR

Group

Fetch commands

Syntax

`FETCh:LTE:ACLR:RESUlts:STATus?`

Returns

`0`: means measurement status is FAIL.

`1`: means measurement status is PASS.

Examples

`FETCH:LTE:ACLR:RESULTS:STATUS?` might return `1`, indicating that the LTE ACLR measurement status is PASS.

FETCh:LTE:ACLR:SPECtrum:X? (Query Only)

Returns the frequencies of the spectrum trace in the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

Group

Fetch commands

Syntax

`FETCh:LTE:ACLR:SPECtrum:X?`

Returns

#<num_digit><num_byte><x(1)><x(2)>...<x(n)>

Where,

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<x(n)> is the frequency (Hz) at the n^{th} point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

FETCH:LTE:ACLR:SPECTRUM:X? might return #516020xxxx... (16020-byte data) for the frequencies of the spectrum trace.

FETCh:LTE:ACLR:SPECtrum:XY? (Query Only)

Returns the frequency and amplitude pairs of the spectrum trace in the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

Group

Fetch commands

Syntax

FETCh:LTE:ACLR:SPECtrum:XY?

Returns

#<num_digit><num_byte><x(1)><y(1)><x(2)><y(2)>...<x(n)><y(n)>

Where,

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<x(n)><y(n)> is the frequency (Hz) and amplitude (dBm) pair at the *nth* point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

FETCH:LTE:ACLR:SPECTRUM:XY? might return #516020xxxx... (16020-byte data) for the frequency and amplitude pairs of the spectrum trace.

FETCh:LTE:ACLR:SPECtrum:Y? (Query Only)

Returns the amplitude of the spectrum trace in the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

Group

Fetch commands

Syntax

FETCh:LTE:ACLR:SPECtrum:Y?

Returns

`#<num_digit><num_byte><y(1)><y(2)>...<y(n)>`

Where,

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<y(n)>` is the amplitude (dB) at the *nth* point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`FETCH:LTE:ACLR:SPECTRUM:Y?` might return `#516020xxxx...` (16020-byte data) for the amplitude of the spectrum trace.

FETCh:LTE:CHSPectrum:OBW? (Query Only)

Returns the Occupied Bandwidth (OBW) in Hz of the LTE Channel Spectrum measurement view.

Conditions

Measurement view: LTE Channel Spectrum

Group

Fetch commands

Syntax

`FETCh:LTE:CHSPectrum:OBW?`

Returns

<NR3>

Examples

`FETCH:LTE:CHSPECTRUM:OBW?` might return `1.0851851250E+6`, indicating the Occupied Bandwidth is 1.09 MHz.

FETCh:LTE:CHSPectrum:POWer:CHANnel? (Query Only)

Returns the Channel Power value in LTE Channel Spectrum display. The result will be in dBm (by default) or in the units chosen from the Units tab of the Analysis Control panel.

Conditions

Measurement view: LTE Channel Spectrum

Group

Fetch commands

Syntax

`FETCh:LTE:CHSPectrum:POWer:CHANnel?`

Returns

<NRf>

Examples

FETCH:LTE:CHSPECTRUM:POWER:CHANNEL? might return –14.9248560147, indicating that the Channel Power is -14.92 dBm.

FETCh:LTE:CHSPectrum:SPECtrum? (Query Only)

Returns spectrum trace data of the LTE Channel Spectrum measurement.

Conditions

Measurement view: LTE Channel Spectrum

Group

Fetch commands

Syntax

FETCh:LTE:CHSPectrum:SPECtrum?

Returns

#<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the amplitude in dBm at the n^{th} data point, 4-byte little endian floating point format specified in IEEE 488.2.

Examples

FETCH:LTE:CHSPECTRUM:SPECTRUM? might return #43204 xxxx... (3204-byte data) for the spectrum trace data of the LTE Channel Spectrum measurement.

FETCh:LTE:CONSte:CELL:ID? (Query Only)

Queries the Cell ID value LTE Constellation display.

Conditions

Measurement view: LTE Constellation

Group

Fetch commands

Syntax

FETCh:LTE:CONSte:CELL:ID?

Returns

<NR1>

Examples

FETCH:LTE:CONSTE:CELL:ID? might return 112, indicating CELLID measurement number is 112.

FETCh:LTE:CONSte:FREQuency:ERRor? (Query Only)

Queries Frequency Error value in Hz LTE Constellation display.

Conditions

Measurement view: LTE Constellation

Group

Fetch commands

Syntax

`FETCh:LTE:CONSte:FREQuency:ERRor?`

Returns

<NRf>

Examples

`FETCH:LTE:CONSTE:FREQUENCY:ERROR?` might return `-71.9780578613`, indicating frequency error value is -71.9780578613 Hz.

FETCh:LTE:CONSte:GROUp:ID? (Query Only)

Queries the Group ID value LTE Constellation display.

Conditions

Measurement view: LTE Constellation

Group

Fetch commands

Syntax

`FETCh:LTE:CONSte:GROUp:ID?`

Returns

<NR1>

Examples

`FETCH:LTE:CONSTE:GROUP:ID?` might return `160`, indicating Group ID measurement number is 160.

FETCh:LTE:CONSte:SECTor:ID? (Query Only)

Queries the Sector ID value for the LTE Constellation display.

Conditions

Measurement view: LTE Constellation

Group

Fetch commands

Syntax

`FETCh:LTE:CONSte:SECTor:ID?`

Returns

<NRf>

Examples

`FETCH:LTE:CONSTE:SECTOR:ID?` might return `2`, indicating the Sector ID value is 2.

FETCh:LTE:PVTime:OFFSlot:POWer? (Query Only)

Returns the TOff power measurement value in dBm/MHz for the LTE Power vs Time display.

Conditions

Measurement view: LTE Power vs Time

Group

Fetch commands

Syntax

`FETCh:LTE:PVTime:OFFSlot:POWer?`

Returns

<NRf>

Examples

`FETCH:LTE:PVTIME:OFFSLOT:POWER?` might return `-76.11514587403`, indicating the offslot power is -76.11514 dBm/MHz.

FETCh:LTE:PVTime:RESUlts:STATus

Returns the pass or fail status for the LTE Power vs Time display.

Conditions

Measurement view: LTE Power vs Time

Group

Fetch commands

Syntax

`FETCh:LTE:PVTime:RESUlts:STATus`

Returns

`0`: means measurement status is FAIL.

`1`: means measurement status is PASS.

Examples

`FETCH:LTE:PVTIME:RESULTS:STATUS` might return `1`, indicating the measurement status is PASS.

FETCh:LTE:PVTime:TRACe:X? (Query Only)

Returns the horizontal values (time in seconds) for the LTE Power vs. Time trace.

Conditions

Measurement view: LTE Power vs Time

Group

Fetch commands

Syntax

`FETCh:LTE:PVTime:TRACe:X?`

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where,

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the *n*th point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`FETCH:LTE:PVTIME:TRACE:X?` might return `#43204` (3204-byte data), which represents the time in seconds (horizontal values).

FETCh:LTE:PVTime:TRACe:XY? (Query Only)

Returns the horizontal value (time in seconds) and vertical value (power) for the LTE Power vs. Time trace.

Conditions

Measurement view: LTE Power vs Time

Group

Fetch commands

Syntax

`FETCh:LTE:PVTime:TRACe:XY?`

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where,

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the *n*th point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`FETCH:LTE:PVTIME:TRACE:XY?` might return `#43204` (3204-byte data), which represents the horizontal value (time in seconds) and vertical value (power) pair at the *n*th data point.

FETCh:LTE:PVTime:TRACe:Y? (Query Only)

Returns the vertical values (power) for the LTE Power vs. Time trace.

Conditions

Measurement view: LTE Power vs Time

Group

Fetch commands

Syntax

```
FETCh:LTE:PVTime:TRACe:Y?
```

Returns

```
#<num_digit><num_byte><data(1)><data(2)>...<data(n)>
```

Where,

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the n^{th} point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`FETCH:LTE:PVTIME:TRACE:Y?` might return `#43204` (3204-byte data), which represents the vertical values (power).

FETCh:MCPower:ADJacent:CHANnels? (Query Only)

Returns the power of adjacent channels in order of increasing frequency.

Conditions

Measurement views: MCPR

Group

Fetch commands

Syntax

```
FETCh:MCPower:ADJacent:CHANnels?
```

Arguments

None

Returns

```
<acpr_lower(n)>,...<acpr_lower(2)>,<acpr_lower(1)>,
<acpr_upper(1)>,<acpr_upper(2)>,...<acpr_upper(n)>
```

Where

`<acpr_lower(n)>` is the ACPR for the lower channel #n in dB.

`<acpr_upper(n)>` is the ACPR for the upper channel #n in dB.

To add a pair of upper and lower adjacent channels, use the `[SENSe]:MCPower: CHANnel:ADJacent:ADD` command.

Examples

`FETCH:MCPOWER:ADJACENT:CHANNELS?` might return `-4.420,-4.847,-4.316,-4.225`, indicating (ACPR for the lower channel 2) = -4.420 dB, (ACPR for the lower channel 1) = -4.847 dB, (ACPR for the upper channel 1) = -4.316 dB, and (ACPR for the upper channel 2) = -4.225 dB.

FETCh:MCPower:CHANnel:POWer? (Query Only)

Returns the reference power in the MCPR measurement.

Conditions

Measurement views: MCPR

Group

Fetch commands

Syntax

`FETCh:MCPower:CHANnel:POWer?`

Arguments

None

Returns

`<ref_power>:<NRf>` is the reference power in dBm.

The unit can be changed by the `[SENSe]:POWer:UNITs` command.

To select the power reference, use the `[SENSe]:MCPower:RCHannels?` commands.

Examples

`FETCH:MCPOWER:CHANNEL:POWER?` might return `4.227`, indicating that the reference power is 4.227 dBm.

FETCh:MCPower:MAIN:CHANnels? (Query Only)

Returns the power of main channels in order of increasing frequency.

Conditions

Measurement views: MCPR

Group

Fetch commands

Syntax

`FETCh:MCPower:MAIN:CHANnels?`

Arguments

None

Returns

`<power_main(1)>,<power_main(2)>,...<power_main(n)>`

Where

`<power_main(n)>` is the power of main channel #n in dBm.

The unit can be changed by the `[SENSe]:POWer:UNITs` command.

To specify the main channels, use the `[SENSe]:MCPower:CHANnel:MAIN` commands.

Examples

`FETCH:MCPOWER:MAIN:CHANNELS?` might return `-2.420,-2.847,-2.316,-2.225`, indicating (power of the main channel 1) = -2.420 dBm, (power of the main channel 2) = -2.847 dBm, (power of the main channel 3) = -2.316 dBm, and (power of the main channel 4) = -2.225 dBm.

FETCh:MCPower:SPECtrum? (Query Only)

Returns spectrum trace data of the MCPR measurement.

Conditions

Measurement views: MCPR

Group

Fetch commands

Syntax

`FETCh:MCPower:SPECtrum?`

Arguments

None

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the amplitude in dBm at the nth data point,

4-byte little endian floating-point format specified in IEEE 488.2. The unit can be changed by the `[SENSe]:POWer:UNITs` command.

Examples

`FETCH:MCPOWER:SPECTRUM?` might return `#43204xxxx...` (3204-byte data) for the spectrum trace data of the MCPR measurement.

FETCh:MERRor[:MEASview<y>]:FERRor? (Query only)

Returns the frequency error in the Magnitude error versus Time measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Magnitude error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Magnitude error versus Time).

Group

Fetch commands

Syntax

`FETCh:MERRor[:MEASview<y>]:FERRor?`

Arguments

None

Returns

<freq_error>::<NRf> is the frequency error in Hz.

Examples

FETCh:MERRor:MEASview2:FERRor? might return -10.7E+3, indicating the frequency error is -10.7 kHz.

FETCh:MERRor[:MEASview<y>]:PEAK? (Query only)

Returns the peak value in the Magnitude error versus Time measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Magnitude error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Magnitude error versus Time).

Group

Fetch commands

Syntax

FETCh:MERRor[:MEASview<y>]:PEAK?

Related Commands

FETCh:MERRor[:MEASview<y>]:PINDex? (on page 1191)

Arguments

None

Returns

<peak>::<NRf> is the peak magnitude error value in percent (%).

Examples

FETCh:MERRor:MEASview2:PEAK? might return 1.57, indicating the peak magnitude error value is 1.57%.

FETCh:MERRor[:MEASview<y>]:PINDex? (Query only)

Returns the time at the magnitude error peak.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Magnitude error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Magnitude error versus Time).

Group

Fetch commands

Syntax

```
FETCh:MERRor[:MEASview<y>]:PINDex?
```

Related Commands

FETCh:MERRor[:MEASview<y>]:PEAK? (on page 1191)

Arguments

None

Returns

`<peak_time>::<NRf>` is the time at the magnitude error peak in symbol number.

The unit can be changed by the `[SENSe]:DDEMod:TIMe:UNITs` command.

Examples

`FETCh:MERRor:MEASview2:PINDex?` might return 68.000, indicating that the Magnitude error peak is at symbol #68.

FETCh:MERRor[:MEASview<y>]:RMS? (Query only)

Returns the RMS (Root-Mean-Square) value in the Magnitude error versus Time measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Magnitude error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Magnitude error versus Time).

Group

Fetch commands

Syntax

```
FETCh:MERRor[:MEASview<y>]:RMS?
```

Arguments

None

Returns

`<rms>::<NRf>` is the RMS magnitude error in percent (%).

Examples

`FETCh:MERRor:MEASview2:RMS?` might return 0.382, indicating the RMS Magnitude error value is 0.382%.

FETCh:MERRor[:MEASview<y>]:TRACe? (Query only)

Returns the Magnitude error versus Time trace data.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Magnitude error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Magnitude error versus Time).

Group

Fetch commands

Syntax

`FETCh:MERRor[:MEASview<y>]:TRACe?`

Arguments

None

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the magnitude error in percent (%) at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`FETCh:MERRor:MEASview2:TRACe?` might return #42036xxxx... (2036-byte data) for the Magnitude error versus Time Measview 2.

FETCh:NOISe:GAIN:TRACe<x>:RESult? (Query Only)

This query returns the gain value in mdB of the specified trace in the Gain display.

Conditions

Measurement view: Gain

The parameter <x> = 1 to 3.

Group

Fetch commands

Syntax

`FETCh:NOISe:GAIN:TRACe<x>:RESult?`

Returns

The gain value in mdB of the specified trace.

Examples

`FETCH:NOISE:GAIN:TRACE1:RESULT?` might return `-88.7`, indicating that the gain is -88.7 mdB.

FETCh:NOISe:TABLe:VALue? (Query Only)

Returns Noise Table data. A total of 7 values per frequency (28 bytes per frequency) is returned. The number of frequencies returned depends on the settings.

Conditions

Measurement view: Noise Table

Group

Fetch commands

Syntax

`FETCh:NOISe:TABLe:VALue?`

Returns

Returns an array of floats (size of 4 bytes each) defined as:

- First frequency gain, noise figure, noise factor, y factor, noise temperature, power hot, power cold
- Second frequency gain, noise figure, noise factor, y factor, noise temperature, power hot, power cold

Examples

`FETCH:NOISE:TABLE:VALUE?` might return #3308, indicating that the

FETCh:NR:ACP:TAGGpower ? (Query Only)

Queries the Total Aggregate Power of the NR ACP display in the NR Summary View table.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax

`FETCh:NR:ACP:TAGGpower?`

Returns

<NRf>

Examples

`FETCH:NR:ACP:TAGGPOWER?` might return `-18.4524345398 dBm`, indicating the Total Aggregate Power.

FETCh:NR:ACP:NO:OF:NR:OFFsets? (Query Only)

Queries NR Offsets of the NR ACP display in the NR Summary View Table.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax

`FETCh:NR:ACP:NO:OF:NR:OFFsets?`

Returns

<NRf>

Examples

`FETCH:NR:ACP:NO:OF:NR:OFFSETS?` might return `6.0000000000`, indicating the maximum offset is 6.

FETCh:NR:ACP:ADJChannel<x>:INTBw? (Query Only)

Queries the bandwidth of the adjacent channel of the NR ACP in NR Summary View Table. <x> is the adjacent channel. Range is 1 to 6.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax

`FETCh:NR:ACP:ADJChannel<x>:INTBw?`

Returns

<NRf>

Examples

`FETCh:NR:ACP:ADJChannel2:INTBw?` might return `4.5150000000E+6 Hz`, indicating the bandwidth of adjacent channel 2.

FETCh:NR:ACP:ADJChannel<x>:UABS? (Query Only)

Queries the upper absolute value of the adjacent channel of the NR ACP in NR Summary View Table. <x> is the adjacent channel. Range is 1 to 6.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax

`FETCh:NR:ACP:ADJChannel<x>:UABS?`

Returns

<NRf>

Examples

`FETCh:NR:ACP:ADJChannel2:UABS?` might return `-46.5615577698 dB`, indicating the upper absolute value of adjacent channel 2.

FETCh:NR:ACP:ADJChannel<x>:UREL ? (Query Only)

Queries the upper relative value of the adjacent channel of the NR ACP in NR Summary View Table. <x> is the adjacent channel. Range is 1 to 6.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax

`FETCh:NR:ACP:ADJChannel<x>:UREL?`

Returns

<NRf>

Examples

`FETCh:NR:ACP:ADJChannel1:UREL?` might return `-47.5079994202 dB`, indicating the upper relative value of adjacent channel 1.

FETCh:NR:ACP:ADJChannel<x>:LABS? (Query Only)

Queries the lower absolute value of the adjacent channel of the NR ACP in NR Summary View Table. <x> is the adjacent channel. Range is 1 to 6.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax

`FETCh:NR:ACP:ADJChannel<x>:LABS?`

Returns

<NRf>

Examples

`FETCh:NR:ACP:ADJChannel1:LABS?` might return `-65.0139923096 dB`, indicating the lower absolute value of adjacent channel 1.

FETCh:NR:ACP:ADJChannel<x>:LREL ? (Query Only)

Queries the lower relative value of the adjacent channel of the NR ACP in NR Summary View Table. <x> is the adjacent channel. Range is 1 to 6.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax

`FETCh:NR:ACP:ADJChannel<x>:LREL?`

Returns

<NRf>

Examples

`FETCh:NR:ACP:ADJChannel2:LREL?` might return `-46.5615577698 dB`, indicating the lower relative value of adjacent channel 2.

FETCh:NR:ACP:ADJChannel<x>:FREQoffset ? (Query Only)

Queries the frequency offset of the adjacent channel of the NR ACP in the NR Summary View Table. <x> is the adjacent channel. Range is 1 to 6.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax

`FETCh:NR:ACP:ADJChannel<x>:FREQoffset?`

Returns

<NRf>

Examples

`FETCh:NR:ACP:ADJChannel2:FREQoffset?` might return `5.0000000000E+6 Hz`, indicating the frequency offset of adjacent channel 2.

FETCh:NR:CHP:CARRier<x>:ABSPower? (Query Only)

Queries the absolute power of each component carrier in the NR Channel Power in NR Summary View Table. <x> is the component carrier. Range is 1 to 8.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax

`FETCh:NR:CHP:CARRier<x>:ABSPower?`

Returns

<NRf>

Examples

`FETCh:NR:CHP:CARRier6:ABSPower?` might return `-13.8919143677 dBm`, indicating the absolute power of component carrier 6.

FETCh:NR:CHP:CARRier<x>:RELPower? (Query Only)

Queries the relative power of each component carrier in the NR Channel Power in NR Summary View Table. <x> is the component carrier. Range is 1 to 8.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax

`FETCh:NR:CHP:CARRier<x>:RELPower?`

Returns

<NRf>

Examples

`FETCh:NR:CHP:CARRier3:RELPower?` might return `0.0000`, indicating the relative power of component carrier 3.

FETCh:NR:CHP:TAGGpower? (Query Only)

Queries the total aggregate power of the NR Channel Power in the NR Summary View Table.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax

`FETCh:NR:CHP:TAGGpower?`

Returns

<NRf>

Examples

`FETCH:NR:CHP:TAGGPOWER?` might return `-13.8919143677 dBm`, indicating the total aggregate power.

FETCh:NR:EVM:ATSYmbol (Query Only)

Queries Symbol value for the selected trace in NR **EVM**.

Conditions

Measurement view: NR EVM measurement view

Group

Fetch commands

Syntax

`FETCh:NR:EVM:ATSYmbol`

Returns

<NRf>

Examples

FETCh:NR:EVM:ATSYmbol? might return 9.9988253091E-3% indicating Symbol value for the selected trace.

FETCh:NR:EVM:PEAK (Query Only)

Queries Peak value for the selected trace in NR **EVM**.

Conditions

Measurement view: NR EVM measurement view

Group

Fetch commands

Syntax

FETCh:NR:EVM:PEAK

Returns

<NRf>

Examples

FETCh:NR:EVM:PEAK? might return 9.9988253091E-3% indicating Peak value for the selected trace.

FETCh:NR:EVM:RMS (Query Only)

Queries RMS value for the selected trace in NR **EVM**.

Conditions

Measurement view: NR EVM measurement view

Group

Fetch commands

Syntax

FETCh:NR:EVM:RMS

Returns

<NRf>

Examples

FETCh:NR:EVM:RMS? might return 9.9988253091E-3% indicating RMS value for the selected trace.

FETCh:NR:LINK:DIREction? (Query Only)

Queries the link direction of the NR Constellation in NR Summary View Table.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax

`FETCh:NR:LINK:DIREction?`

Returns

<NRf>

Examples

`FETCH:NR:LINK:DIRECTION?` might return `DOWN`, indicating that the link direction is downlink.

FETCh:NR:MODAcc:CARRier<x>:CELL:ID? (Query Only)

Queries the Cell ID of the NR Constellation in NR Summary View Table. <x> is the component carrier. Range is 1 to 8.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax

`FETCh:NR:MODAcc:CARRier<x>:CELL:ID?`

Returns

<NRf>

Examples

`FETCh:NR:MODAcc:CARRier5:CELL:ID?` might return `1`, indicating that the cell ID is 5.

FETCh:NR:MODAcc:CARRier<x>:OFFSet? (Query Only)

Queries the carrier offset of the NR Constellation in NR Summary View Table.

<x> is the component carrier. Range is 1 to 8.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax

`FETCh:NR:MODAcc:CARRier<x>:OFFSet?`

Returns

<NRf>

Examples

`FETCh:NR:MODAcc:CARRier2:OFFSet?` might return `-184.740`, indicating the offset value for component carrier 2 is -184.740 MHz.

FETCh:NR:MODAcc:CARRier<x>:CREM? (Query Only)

Queries the Composite RMS **EVM** Mean Value of the NR Constellation in NR Summary View Table. <x> is the component carrier. Range is 1 to 8.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax

`FETCh:NR:MODAcc:CARRier<x>:CREM?`

Returns

<NRf>

Examples

`FETCh:NR:MODAcc:CARRier2:CREM?` might return `255.7628555223E-3`, indicating that the component carrier 2 Composite RMS EVM Mean value is 0.256%.

FETCh:NR:MODAcc:CARRier<x>:CPEM? (Query Only)

Queries the Composite Peak **EVM** Maximum Value of NR Constellation in NR Summary View Table. <x> is the component carrier. Range is 1 to 8.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax

`FETCh:NR:MODAcc:CARRier<x>:CPEM?`

Returns

<NRf>

Examples

`FETCh:NR:MODAcc:CARRier3:CPEM?` might return `672.3846774548E-3`, indicating the component carrier 3 Composite Peak EVM Maximum Value is 0.672%.

FETCh:NR:MODAcc:CARRier<x>:CPESLI? (Query Only)

Queries the Composite Peak Slot Index of NR Constellation in NR Summary View Table. <x> is the component carrier. Range is 1 to 8.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax

`FETCh:NR:MODAcc:CARRier<x>:CPESLI?`

Returns

<NRf>

Examples

`FETCh:NR:MODAcc:CARRier1:CPESLI?` might return `0`, indicating the component carrier 1 Composite Peak **EVM** Slot Index is 0.

FETCh:NR:MODAcc:CARRier<x>:CPESYI? (Query Only)

Queries the Composite Peak Symbol Index of NR Constellation in NR Summary View Table. <x> is the component carrier. Range is 1 to 8.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax

`FETCh:NR:MODAcc:CARRier<x>:CPESYI?`

Returns

<NRf>

Examples

`FETCh:NR:MODAcc:CARRier2:CPESYI` might return `13`, indicating component carrier 2 Composite Peak **EVM** Symbol Index is 13.

FETCh:NR:MODAcc:CARRier<x>:CPESCI? (Query Only)

Queries the Composite Peak **EVM** Subcarrier Index of NR Constellation in NR Summary View Table. <x> is the component carrier. Range is 1 to 8.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax

`FETCh:NR:MODAcc:CARRier<x>:CPESCI?`

Returns

<NRf>

Examples

`FETCh:NR:MODAcc:CARRier3:CPESCI?` might return `113`, indicating the component carrier 3 Composite Peak EVM Subcarrier Index is 113.

FETCh:NR:MODAcc:CARRier<x>:CCFME? (Query Only)

Queries the Composite Carrier Frequency Mean Error of NR Constellation in NR Summary View Table. <x> is the component carrier. Range is 1 to 8.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax

`FETCh:NR:MODAcc:CARRier<x>:CCFME?`

Returns

<NRf>

Examples

`FETCh:NR:MODAcc:CARRier1:CCFME?` might return `528.9274902344`, indicating the composite carrier 1 Frequency Mean Error is 528.9274902344 Hz.

FETCh:NR:MODAcc:CARRier<x>:CCIOOM? (Query Only)

Queries the Composite Carrier IQ Origin Offset Mean of NR Constellation in NR Summary View Table. <x> is the component carrier. Range is 1 to 8.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax

`FETCh:NR:MODAcc:CARRier<x>:CCIOOM?`

Returns

<NRf>

Examples

`FETCh:NR:MODAcc:CARRier2:CCIOOM?` might return `-70.8180387744`, indicating composite carrier 2 IQ Origin Offset Mean is -70.8180387744 dBc.

FETCh:NR:MODAcc:CARRier<x>:CCIGIM? (Query Only)

Queries the Composite Carrier IQ Gain Imbalance Mean of NR Constellation in NR Summary View Table. <x> is the component carrier. Range is 1 to 8.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax

`FETCh:NR:MODAcc:CARRier<x>:CCIGIM?`

Returns

<NRf>

Examples

`FETCh:NR:MODAcc:CARRier3:CCIGIM?` might return `0.001`, indicating component carrier 3 Composite Carrier IQ Gain Imbalance Mean is 0.001 dB.

FETCh:NR:MODAcc:CARRier<x>:CCIQEM? (Query Only)

Returns the Composite Carrier IQ Quadrature Error Mean of NR Constellation in NR Summary View Table. <x> is the component carrier. Range is 1 to 8.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax

`FETCh:NR:MODAcc:CARRier<x>:CCIQEM?`

Returns

<NRf>

Examples

`FETCh:NR:MODAcc:CARRier3:CCIQEM?` might return `19.3109996617E-3`, indicating component carrier 3 Composite Carrier IQ Quadrature Error Mean is 19.3109996617E-3.

FETCh:NR:MODAcc:CARRier<x>:IBEM? (Query Only)

Queries the In-Band Emission Range of the NR Constellation in NR Summary View Table. <x> is the component carrier. Range is 1 to 8.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax

`FETCh:NR:MODAcc:CARRier<x>:IBEM?`

Returns

<NRf>

Examples

`FETCh:NR:MODAcc:CARRier1:IBEM?` might return `9.9100E+37`, indicating the component carrier 1 In-Band Emission Range.

FETCh:NR:MODAcc:CARRier<x>:DATA:PEAK (Query Only)

Queries the SCH Data Peak **EVM** Maximum Value of NR Constellation in NR Summary View Table display.

The parameter <x>=1 to 8, representing range of Component Carrier.

Conditions

Measurement view: NR Summary View Table measurement view

Group

Fetch commands

Syntax

`FETCh:NR:MODAcc:CARRier<x>:DATA:PEAK?`

Returns

`<NRf>`

Examples

`FETCh:NR:MODAcc:CARRier1:DATA:PEAK?` might return `672.3846774548E-3`, indicating SCH Data Peak EVM Maximum Value is 0.672%.

FETCh:NR:MODAcc:CARRier<x>:DATA:RMS (Query Only)

Queries the SCH Data RMS **EVM** Mean Value of NR Constellation in NR Summary View Table display.

The parameter <x>=1 to 8, representing range of Component Carrier

Conditions

Measurement view: NR Summary View Table measurement view

Group

Fetch commands

Syntax

`FETCh:NR:MODAcc:CARRier<x>:DATA:RMS?`

Returns

`<NRf>`

Examples

`FETCh:NR:MODAcc:CARRier1:DATA:RMS?` might return `286.3456029445E-3`, indicating SCH Data RMS EVM Mean Value is 0.286%.

FETCh:NR:MODAcc:CARRier<x>:DMRS:PEAK (Query Only)

Queries the SCH DMRS Peak **EVM** Maximum Value of NR Constellation in NR Summary View Table display.

The parameter <x>=1 to 8, representing range of Component Carrier.

Conditions

Measurement view: NR Summary View Table measurement view

Group

Fetch commands

Syntax

`FETCh:NR:MODAcc:CARRier<x>:DMRS:PEAK?`

Returns

`<NRf>`

Examples

`FETCh:NR:MODAcc:CARRier1:DMRS:PEAK?` might return `616.6561972350E-3`, indicating SCH DMRS Peak EVM Maximum Value is 0.617%.

FETCh:NR:MODAcc:CARRier<x>:DMRS:RMS (Query Only)

Queries the SCH DMRS **EVM** Mean Value of NR Constellation in NR Summary View Table display.

The parameter <x>=1 to 8, representing range of Component Carrier.

Conditions

Measurement view: NR Summary View Table measurement view

Group

Fetch commands

Syntax

`FETCh:NR:MODAcc:CARRier<x>:DMRS:RMS?`

Returns

`<NRf>`

Examples

`FETCh:NR:MODAcc:CARRier1:DMRS:RMS?` might return `251.3078739867E-3`, indicating SCH DMRS EVM Value is 0.251%.

FETCh:NR:MODAcc:CARRier<x>:PREM (Query Only)

Queries the Peak RMS EVM Maximum Value of NR Constellation in NR Summary View Table display.

The parameter <x>=1 to 8, representing range of Component Carrier.

Conditions

Measurement view: NR Summary View Table measurement view

Group

Fetch commands

Syntax

```
FETCh:NR:MODAcc:CARRier<x>:PREM?
```

Returns

<NRf>

Examples

`FETCh:NR:MODAcc:CARRier1:PREM?` might return `672.3846774548E-3`, indicating Peak EVM Maximum Value is 0.672%.

FETCh:NR:MODAcc:CARRier<x>:PTRS: RMS (Query Only)

Queries the SCH PTRS EVM Mean Value of NR Constellation in NR Summary View Table display.

The parameter <x>=1 to 8, representing range of Component Carrier

Conditions

Measurement view: NR Summary View Table measurement view

Group

Fetch commands

Syntax

```
FETCh:NR:MODAcc:CARRier<x>:PTRS:RMS?
```

Returns

```
<NRf>
```

Examples

`FETCh:NR:MODAcc:CARRier1:PTRS:RMS?` might return `672.3846774548E-3`, indicating SCH PTRS EVM MEAN Value is 0.672%.

FETCh:NR:MODAcc:CARRier<x>:PTRS:PEAK (Query Only)

Queries the SCH PTRS Peak EVM Maximum Value of NR Constellation in NR Summary View Table display.

The parameter <x>=1 to 8, representing range of Component Carrier.

Conditions

Measurement view: NR Summary View Table measurement view

Group

Fetch commands

Syntax
`FETCh:NR:MODAcc:CARRier<x>:PTRS:PEAK?`

Returns
`<NRf>`

Examples
`FETCh:NR:MODAcc:CARRier1:PTRS:PEAK?` might return `815.3068833053E-3`, indicating SCH PTRS Peak EVM Maximum.

FETCh:NR:OBW:ABSPower (Query Only)
Queries Absolute power of NR OBW display in NR Summary View Table display.

Conditions
Measurement view: NR Summary View Table measurement view

Group
Fetch commands

Syntax
`FETCh:NR:OBW:ABSPower`

Returns
<NRf>

Examples
`FETCh:NR:OBW:ABSPower?` might return `-13.8897142410 dBm`, indicating the Absolute power.

FETCh:NR:OBW:OCCUpiedbw (Query Only)
Queries Occupied Bandwidth of NR OBW display in NR Summary View Table display.

Conditions
Measurement view: NR Summary View Table measurement view

Group
Fetch commands

Syntax
`FETCh:NR:OBW:OCCUpiedbw`

Returns
`<NRf>`

Examples
`FETCh:NR:OBW:OCCUpiedbw?` might return `3.9506140488 MHz`, indicating the Occupied Bandwidth.

FETCh:NR:OBW:STARtfreq (Query Only)

Queries Start frequency of NR OBW display in NR Summary View Table display.

Conditions

Measurement view: NR Summary View Table measurement view

Group

Fetch commands

Syntax

```
FETCh:NR:OBW:STARtfreq
```

Returns

<NRf>

Examples

`FETCh:NR:OBW:STARtfreq?` might return `998.0183831384E+6`, indicating the Stop frequency.

FETCh:NR:OBW:STOPfreq (Query Only)

Queries Stop frequency of NR OBW display in NR Summary View Table display.

Conditions

Measurement view: NR Summary View Table measurement view

Group

Fetch commands

Syntax

```
FETCh:NR:OBW:STOPfreq
```

Returns

<NRf>

Examples

`FETCh:NR:OBW:STOPfreq?` might return `1.0019689972E+9`, indicating the Stop frequency.

FETCh:NR:PVT:ABS:OFFpower:AFTR (Query Only)

Queries Absolute Off Power After NR Power vs Time display in NR Summary View Table display.

Conditions

Measurement view: NR Summary View Table measurement view

Group

Fetch commands

Syntax

```
FETCh:NR:PVT:ABS:OFFpower:AFTR
```

Returns

<NRf>

Examples

FETCh:NR:PVT:ABS:OFFpower:AFTR? might return -62.4365692139 dBm, indicating the Absolute Off Power After.

FETCh:NR:PVT:ABS:OFFpower:BFR (Query Only)

Queries Absolute Off Power Before of NR Power vs Time display in NR Summary View Table display.

Conditions

Measurement view: NR Summary View Table measurement view

Group

Fetch commands

Syntax

FETCh:NR:PVT:ABS:OFFpower:BFR

Returns

<NRf>

Examples

FETCh:NR:PVT:ABS:OFFpower:BFR? might return -62.4902038574 dBm, indicating the Absolute Off Power Before.

FETCh:NR:PVT:ABS:ONpower (Query Only)

Queries Absolute ON Power NR Power vs Time display in NR Summary View Table display.

Conditions

Measurement view: NR Summary View Table measurement view

Group

Fetch commands

Syntax

FETCh:NR:PVT:ABS:ONpower

Returns

<NRf>

Examples

FETCh:NR:PVT:ABS:ONpower? might return -22.0409736633 dBm, indicating Absolute on power.

FETCh:NR:PVT:BURStwidth (Query Only)

Queries Burst Width NR Power vs Time display in NR Summary View Table display.

Conditions

Measurement view: NR Summary View Table measurement view

Group

Fetch commands

Syntax

`FETCh:NR:PVT:BURStwidth`

Returns

`<NRf>`

Examples

`FETCh:NR:PVT:BURStwidth?` might return `9.9988253091E-3 s`, indicating Burst Width.

FETCh:NR:PVT:MEASurement:STATus (Query Only)

Queries measurement status of NR Power vs Time display in NR Summary View Table display.

Conditions

Measurement view: NR Summary View Table measurement view

Group

Fetch commands

Syntax

`FETCh:NR:PVT:MEASurement:STATus`

Returns

`<NRf>`

Examples

`FETCh:NR:PVT:MEASurement:STATus?` might return `"PASS"`, indicating the Power vs Time passed.

FETCh:NR:SEM:TAGGpower? (Query Only)

Queries the Total Aggregate Power of the NR SEM in NR Summary View Table.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax

`FETCh:NR:SEM:TAGGpower?`

Returns

<NRf>

Examples

`FETCH:NR:SEM:TAGGPOWER?` might return -14.0847625732 dBm, indicating the Total Aggregate Power.

FETCh:NR:SEM:OFFSet<x>:STARtfreq? (Query Only)

Queries the Start Frequency of the specified offset of the NR SEM display in NR Summary View Table. <x> is the offset. Range is 1 to 16.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax

`FETCh:NR:SEM:OFFSet<x>:STARtfreq?`

Returns

<NRf>

Examples

`FETCh:NR:SEM:OFFSet5:STARtfreq?` might return -25.0000000000E+3 Hz, indicating the Start Frequency of offset 5.

FETCh:NR:SEM:OFFSet<x>:STOPfreq? (Query Only)

Queries the Stop Frequency of the specified offset of the NR SEM display in NR Summary View Table. <x> is the offset. Range is 1 to 16.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax

`FETCh:NR:SEM:OFFSet<x>:STOPfreq?`

Returns

<NRf>

Examples

`FETCh:NR:SEM:OFFSet10:STOPtfreq?` might return `-975.0000000000E+3 Hz`, indicating the Stop Frequency of offset 10.

FETCh:NR:SEM:OFFSet<x>:PEAK:ABS? (Query Only)

Queries the peak absolute value of specified offset of NR SEM display in NR Summary View Table. <x> is the offset. Range is 1 to 16.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax
`FETCh:NR:SEM:OFFSet<x>:PEAK:ABS?`

Returns
<NRf>

Examples
`FETCh:NR:SEM:OFFSet1:PEAK:ABS?` might return `-65.5201115608 dBm`, indicating the Peak Absolute value of offset 1.

FETCh:NR:SEM:OFFSet<x>:PEAK:REL? (Query Only)
Queries the peak relative value of the specified offset of the NR SEM display in the NR Summary View Table. <x> is the offset. Range is 1 to 16.

Conditions
Measurement view: NR Summary View Table

Group
Fetch commands

Syntax
`FETCh:NR:SEM:OFFSet<x>:PEAK:REL?`

Returns
<NRf>

Examples
`FETCh:NR:SEM:OFFSet1:PEAK:REL?` might return -51.4353489876 dB, indicating the Peak Relative of offset 1.

FETCh:NR:SEM:OFFSet<x>:PEAK:FREQ? (Query Only)
Queries the peak frequency of the specified offset of the NR SEM display in the NR Summary View Table. <x> is the offset. Range is 1 to 16.

Conditions
Measurement view: NR Summary View Table

Group
Fetch commands

Syntax
`FETCh:NR:SEM:OFFSet<x>:PEAK:FREQ?`

Returns
<NRf>

Examples
`FETCh:NR:SEM:OFFSet1:PEAK:FREQ?` might return `997.4753714650E+6 GHz`, indicating the Peak Frequency of offset 1.

FETCh:NR:SEM:OFFSet<x>:MARGin? (Query Only)

Queries the margin of the specified offset of the NR SEM display in the NR Summary View Table. <x> is the offset. Range is 1 to 16.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax

`FETCh:NR:SEM:OFFSet<x>:MARGin?`

Returns

<NRf>

Examples

`FETCh:NR:SEM:OFFSet3:MARGin?` might return `-52.5201110840 dB`, indicating the Margin of offset 3.

FETCh:NR:SEM:OFFSet<x>:MARGin:ABS? (Query Only)

Queries the margin absolute value of the specified offset of the NR SEM display in the NR Summary View Table. <x> is the offset. Range is 1 to 16.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax

`FETCh:NR:SEM:OFFSet<x>:MARGin:ABS?`

Returns

<NRf>

Examples

`FETCh:NR:SEM:OFFSet2:MARGin:ABS?` might return `-51.4353489876 dBm`, indicating the Margin Absolute value of offset 2.

FETCh:NR:SEM:OFFSet<x>:MARGin:REL? (Query Only)

Queries the margin relative value of the specified offset of the NR SEM display in the NR Summary View Table. <x> is the offset. Range is 1 to 16.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax
`FETCh:NR:SEM:OFFSet<x>:MARGin:REL?`

Returns
<NRf>

Examples
`FETCh:NR:SEM:OFFSet1:MARGin:REL?` might return `-51.4353489876 dB`, indicating the Margin Relative value of offset 1.

FETCh:NR:SEM:OFFSet<x>:MARGin:FREQ? (Query Only)
Queries the margin frequency of the specified offset of the NR SEM display in the NR Summary View Table <x> is the offset. Range is 1 to 16.

Conditions
Measurement view: NR Summary View Table

Group
Fetch commands

Syntax
`FETCh:NR:SEM:OFFSet<x>:MARGin:FREQ?`

Returns
<NRf>

Examples
`FETCh:NR:SEM:OFFSet1:MARGin:FREQ?` might return `-51.4353489876 GHz`, indicating the margin frequency of offset 1.

FETCh:NR:SEM:OFFSet<x>:INTEg:ABS? (Query Only)
Queries the Integ Absolute value of the specified offset of the NR SEM display in the NR Summary View Table. <x> is the offset. Range is 1 to 16.

Conditions
Measurement view: NR Summary View Table

Group
Fetch commands

Syntax
`FETCh:NR:SEM:OFFSet<x>:INTEg:ABS?`

Returns
<NRf>

Examples
`FETCh:NR:SEM:OFFSet3:INTEg:ABS?` might return `-51.4353489876 dBm`, indicating the Integ Absolute value of offset 3.

FETCh:NR:SEM:OFFSet<x>:INTEg:REL? (Query Only)

Queries the Integ Relative value of the specified offset of the NR SEM display in the NR Summary View Table. <x> is the offset. Range is 1 to 16.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax

`FETCh:NR:SEM:OFFSet<x>:INTEg:REL?`

Returns

<NRf>

Examples

`FETCh:NR:SEM:OFFSet2:INTEg:REL?` might return `-51.4353489876 dB`, indicating the Integ Relative value of offset 2.

FETCh:NR:SEM:OFFSet<x>:RBW? (Query Only)

Queries the RBW of the specified offset of the NR SEM display in the NR Summary View Table. <x> is the offset. Range is 1 to 16.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax

`FETCh:NR:SEM:OFFSet<x>:RBW?`

Returns

<NRf>

Examples

`FETCh:NR:SEM:OFFSet1:RBW?` might return `-51.4353489876 kHz`, indicating the RBW of offset 1.

FETCh:NR:SEM:OFFSet<x>:ABSLimit:STARt? (Query Only)

Queries the Absolute Start value of the specified offset of the NR SEM display in the NR Summary View Table. <x> is the offset. Range is 1 to 16.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax

`FETCh:NR:SEM:OFFSet<x>:ABSLimit:STARt?`

Returns

<NRf>

Examples

`FETCh:NR:SEM:OFFSet2:ABSLimit:STARt?` might return `-51.4353489876 dBm`, indicating the Absolute Start of offset 2.

FETCh:NR:SEM:OFFSet<x>:ABSLimit:STOP? (Query Only)

Queries the Absolute Stop value of the specified offset of the NR SEM display in the NR Summary View Table. <x> is the offset. Range is 1 to 16.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax

`FETCh:NR:SEM:OFFSet<x>:ABSLimit:STOP?`

Returns

<NRf>

Examples

`FETCh:NR:SEM:OFFSet2:ABSLimit:STARt?` might return `-51.4353489876 dBm`, indicating the Absolute Stop value of offset 2.

FETCh:NR:SEM:OFFSet<x>:RELLimit:STARt? (Query Only)

Queries the Relative Start value of the specified offset of the NR SEM display in the NR Summary View Table. <x> is the offset. Range is 1 to 16.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax

`FETCh:NR:SEM:OFFSet<x>:RELLimit:STARt?`

Returns

<NRf>

Examples

`FETCh:NR:SEM:OFFSet2:RELLimit:STARt?` might return `-51.4353489876 dBm`, indicating the Relative Start value of offset 2.

FETCh:NR:SEM:OFFSet<x>:RELLimit:STOP? (Query Only)

Queries the Relative Stop value of the specified offset of the NR SEM display in the NR Summary View Table. <x> is the offset. Range is 1 to 16.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax

`FETCh:NR:SEM:OFFSet<x>:RELLimit:STOP?`

Returns

<NRf>

Examples

`FETCh:NR:SEM:OFFSet2:RELLimit:STOP?` might return `-51.4353489876 dBm`, indicating the Relative Stop value of offset 2.

FETCh:NR:SUBblock<x>:NUMCcs? (Query Only)

Queries the number of component carriers in the specified subblock of the NR Constellation in the NR Summary View Table. <x> is the subblock. Range is 1.

Conditions

Measurement view: NR Summary View Table

Group

Fetch commands

Syntax

`FETCh:NR:SUBblock<x>:NUMCcs?`

Returns

<NRf>

Examples

`FETCh:NR:SUBblock1:NUMCcs?` might return `8`, indicating that the number of component carriers in subblock 1 is 8.

FETCh:OBWidth:BOBW:FREQuency:ERRor (Query Only)

Returns the frequency error in the Bluetooth 20dB BW display.

Conditions

Measurement view: Bluetooth 20dB BW

Group

Fetch commands

Syntax
`FETCh:OBWidth:BOBW:FREQuency:ERRor?`

Arguments
None

Returns
<NRf>

Examples
`FETCH:OBWIDTH:BOBW:FREQUENCY:ERROR` might return `-999.1 kHz`, indicating the frequency error is `-999.1 kHz`.

FETCh:OBWidth:BOBW:RESUlts:STATus (Query Only)
Returns the test result as Pass or Fail in the Bluetooth 20dB BW display.

Conditions
Measurement views: Bluetooth 20dB BW

Group
Fetch commands

Syntax
`FETCh:OBWidth:BOBW:RESUlts:STATus`

Arguments
None

Returns
`PASS` means the test passed.

`FAIL` means the test failed.

`INVALID` means there was an error and the test did not complete.

Examples
`FETCH:OBWIDTH:BOBW:RESULTS:STATUS` might return `Pass` or `Fail` or `Invalid` for Bluetooth 20dB BW display.

FETCh:OBWidth:BOBW:XDBB:POWer (Query Only)
Returns the x dB BW Ref Power value for the selected bandwidth as shown in the Bluetooth 20dB BW display.

Conditions
Measurement view: Bluetooth 20dB BW

Group
Fetch commands

Syntax
`FETCh:OBWidth:BOBW:XDBB:POWer?`

Returns

<NR3>

Examples

`FETCH:OBWIDTH:BOBW:XDBB:POWER` might return `-9.0197525024`, indicating that the x dB reference power for the selected bandwidth is -9.0197525024 dB.

FETCh:OBWidth:BOBW:XDBBandwidth? (Query Only)

Returns the difference between the higher and lower frequency points corresponding to the value where it is X dB less from the value at the center frequency.

Set the measurement direction using the appropriate inward or outward command before issuing this command.

Conditions

Measurement views: Bluetooth 20 dB Bandwith

Group

Fetch commands

Syntax

`FETCh:OBWidth:BOBW:XDBBandwidth?`

Related Commands

FETCh:OBWidth:BOBW:XDBMbandwidth:IN:BANDwidth? (on page 1220)

FETCh:OBWidth:BOBW:XDBMbandwidth:OUT:BANDwidth? (on page 1222)

Arguments

None

Returns

<value> ::= <NRf> the X dBm bandwidth in Hz.

Examples

`FETCH:OBWIDTH:BOBW:XDBBANDWIDTH?` might return `89.12000E+3`, indicating the difference between the higher and lower frequency points is 89.12 kHz.

FETCh:OBWidth:BOBW:XDBMbandwidth:IN:BANDwidth? (Query Only)

Returns the difference between the higher and lower frequency points corresponding to the value which it is the X dBm value measured inwards (coming from the ends to the center frequency).

Conditions

Measurement views: Bluetooth 20 dB Bandwith

Group

Fetch commands

Syntax

FETCh:OBWidth:BOBW:XDBMbandwidth:IN:BANDwidth?

Related Commands

FETCh:OBWidth:BOBW:XDBBandwidth? (on page 1220)

FETCh:OBWidth:BOBW:XDBMbandwidth:OUT:BANDwidth? (on page 1222)

Arguments

None

Returns

<value> ::= <NRf> the input X dBm value in Hz.

Examples

FETCH:OBWIDTH:BOBW:XDBMBANDWIDTH:IN:BANDWIDTH? might return 1.2518400000E+6, indicating that the bandwidth corresponding to the input XdBm vaue is 1.2518 MHz measured in the inward direction.

FETCh:OBWidth:BOBW:XDBMbandwidth:IN:LEFT:FREQuency? (Query Only)

Returns the frequency corresponding to the X dBm value to the left of the center frequency measured in the inward direction.

Conditions

Measurement views: Bluetooth 20 dB Bandwith

Group

Fetch commands

Syntax

FETCh:OBWidth:BOBW:XDBMbandwidth:IN:LEFT:FREQuency?

Related Commands

FETCh:OBWidth:BOBW:XDBMbandwidth:IN:RIGHt:FREQuency? (on page 1221)

Arguments

None

Returns

<value> ::= <NRf> the left frequency in Hz.

Examples

FETCH:OBWIDTH:BOBW:XDBMBANDWIDTH:IN:LEFT:FREQUENCY? might return 2.401E+9 indicating that the frequency corresponding to the X dbm value to the left of the center frequency is 2.401 GHz.

FETCh:OBWidth:BOBW:XDBMbandwidth:IN:RIGHt:FREQuency? (Query Only)

Returns the frequency corresponding to the X dBm value to the right of the center frequency measured in the inward direction.

Conditions

Measurement views: Bluetooth 20 dB Bandwith

Group

Fetch commands

Syntax

`FETCh:OBWidth:BOBW:XDBMbandwidth:IN:RIGHt:FREQuency?`

Related Commands

FETCh:OBWidth:BOBW:XDBMbandwidth:IN:LEFT:FREQuency? (on page 1221)

Arguments

None

Returns

<value> ::= <NRf> the right frequency in Hz.

Examples

`FETCH:OBWIDTH:BOBW:XDBMBANDWIDTH:IN:RIGHT:FREQUENCY?` might return `2.445E+9`, indicating that the frequency corresponding to the XdBm value to the right of the center the frequency is 2.445 GHz.

FETCh:OBWidth:BOBW:XDBMbandwidth:OUT:BANDwidth? (Query Only)

Returns the difference between the higher and lower frequency points corresponding to the value which it is the X dBm value measured outwards (from the center frequency to the ends).

Conditions

Measurement views: Bluetooth 20 dB Bandwith

Group

Fetch commands

Syntax

`FETCh:OBWidth:BOBW:XDBMbandwidth:OUT:BANDwidth?`

Related Commands

FETCh:OBWidth:BOBW:XDBBandwidth? (on page 1220)

FETCh:OBWidth:BOBW:XDBMbandwidth:IN:BANDwidth? (on page 1220)

Arguments

None

Returns

<value> ::= <NRf> the input X dBm value in Hz.

Examples

FETCH:OBWIDTH:BOBW:XDBMBANDWIDTH:OUT:BANDWIDTH? might return 1.2518400000E+6, indicating that the bandwidth corresponding to the output XdBm vaue is 1.2518 MHz measured in the outward direction.

FETCh:OBWidth:BOBW:XDBMbandwidth:OUT:LEFT:FREQuency? (Query Only)

Returns the frequency corresponding to the X dBm value to the left of the center frequency measured in the outward direction.

Conditions

Measurement views: Bluetooth 20 dB Bandwith

Group

Fetch commands

Syntax

FETCh:OBWidth:BOBW:XDBMbandwidth:OUT:LEFT:FREQuency?

Related Commands

FETCh:OBWidth:BOBW:XDBMbandwidth:OUT:RIGHt:FREQuency? (on page 1223)

Arguments

None

Returns

<value> ::= <NRf> the left frequency in Hz.

Examples

FETCH:OBWIDTH:BOBW:XDBMBANDWIDTH:OUT:LEFT:FREQUENCY? might return 2.401E+9, indicating that the frequency corresponding to the X dbm value to the left of the center frequency is 2.401 GHz..

FETCh:OBWidth:BOBW:XDBMbandwidth:OUT:RIGHt:FREQuency? (Query Only)

Returns the frequency corresponding to the X dBm value to the right of the center frequency measured in the outward direction.

Conditions

Measurement views: Bluetooth 20 dB Bandwith

Group

Fetch commands

Syntax

FETCh:OBWidth:BOBW:XDBMbandwidth:OUT:RIGHt:FREQuency?

Related Commands

FETCh:OBWidth:BOBW:XDBMbandwidth:OUT:LEFT:FREQuency? (on page 1223)

Arguments

None

Returns

<value> ::= <NRf> the Right frequency in Hz.

Examples

`FETCH:OBWIDTH:BOBW:XDBMBANDWIDTH:OUT:RIGHT:FREQUENCY?` might return `2.445E+9`, indicating that the frequency corresponding to the XdBm value to the right of the center the frequency is 2.445 GHz.

FETCh:OBWidth:FREQuency:ERRor? (Query Only)

Returns the frequency error in the Occupied Bandwidth measurement.

Conditions

Measurement views: Occupied Bandwidth

Group

Fetch commands

Syntax

`FETCh:OBWidth:FREQuency:ERRor?`

Arguments

None

Returns

`<freq_error>::<NRf>` is the frequency error in Hz.

Examples

`FETCH:OBWIDTH:FREQUENCY:ERROR?` might return `-10.7E+3`, indicating the frequency error is -10.7 kHz.

FETCh:OBWidth:OBWidth:BANDwidth? (Query Only)

Returns the occupied bandwidth in the Occupied Bandwidth measurement.

Conditions

Measurement views: Occupied Bandwidth

Group

Fetch commands

Syntax

`FETCh:OBWidth:OBWidth:BANDwidth?`

Arguments

None

Returns

`<OBW>::<NRf>` is the occupied bandwidth in Hz.

Examples

`FETCH:OBWIDTH:OBWIDTH:BANDWIDTH?` might return `4.0E+6`, indicating the occupied bandwidth is 4 MHz.

FETCh:OBWidth:OBWidth:LEFT:FREQuency? (Query Only)

Returns the left (lower) frequency of the occupied bandwidth.

Conditions

Measurement views: Occupied Bandwidth

Group

Fetch commands

Syntax

`FETCh:OBWidth:OBWidth:LEFT:FREQuency?`

Related Commands

FETCh:OBWidth:OBWidth:RIGHt:FREQuency? (on page 1226)

Arguments

None

Returns

`<OBW_left_freq>::<NRf>` is the left frequency in Hz.

Examples

`FETCH:OBWIDTH:OBWIDTH:LEFT:FREQUENCY?` might return `1.498E+9`, indicating the left frequency is 1.498 GHz.

FETCh:OBWidth:OBWidth:LEFT:LEVel? (Query Only)

Returns the level at the left frequency of the occupied bandwidth.

Conditions

Measurement views: Occupied Bandwidth

Group

Fetch commands

Syntax

`FETCh:OBWidth:OBWidth:LEFT:LEVel?`

Related Commands

FETCh:OBWidth:OBWidth:RIGHt:LEVel? (on page 1227)

Arguments

None

Returns

`<OBW_left_level>::<NRf>` is the level at the left frequency in dB.

Examples

`FETCH:OBWIDTH:OBWIDTH:LEFT:LEVEL?` might return `-23.5`, indicating the level at the left frequency is -23.5 dB.

FETCh:OBWidth:OBWidth:POWer? (Query Only)

Returns the reference power in the Occupied Bandwidth measurement.

Conditions

Measurement views: Occupied Bandwidth

Group

Fetch commands

Syntax

`FETCh:OBWidth:OBWidth:POWer?`

Arguments

None

Returns

`<OBW_ref_power>::<NRf>` is the reference power in dBm. The unit can be changed by the `[SENSe]:POWer:UNITs` command.

Examples

`FETCH:OBWIDTH:OBWIDTH:POWER?` might return `-10.0`, indicating the reference power is -10 dBm.

FETCh:OBWidth:OBWidth:RIGHt:FREQuency? (Query Only)

Returns the right (higher) frequency of the occupied bandwidth.

Conditions

Measurement views: Occupied Bandwidth

Group

Fetch commands

Syntax

`FETCh:OBWidth:OBWidth:RIGHt:FREQuency?`

Related Commands

FETCh:OBWidth:OBWidth:LEFT:FREQuency? (on page 1225)

Arguments

None

Returns

`<OBW_right_freq>::<NRf>` is the right frequency in Hz.

Examples

FETCH:OBWIDTH:OBWIDTH:RIGHT:FREQUENCY? might return 1.502E+9, indicating the right frequency is 1.502 GHz.

FETCh:OBWidth:OBWidth:RIGHt:LEVel? (Query Only)

Returns the level at the right frequency of the occupied bandwidth.

Conditions

Measurement views: Occupied Bandwidth

Group

Fetch commands

Syntax

FETCh:OBWidth:OBWidth:RIGHt:LEVel?

Related Commands

FETCh:OBWidth:OBWidth:LEFT:LEVel? (on page 1225)

Arguments

None

Returns

<OBW_right_level>::<NRf> is the level at the right frequency in dB.

Examples

FETCH:OBWIDTH:OBWIDTH:RIGHT:LEVEL? might return -23.5, indicating the level at the right frequency is -23.5 dB.

FETCh:OBWidth:SPECtrum? (Query Only)

Returns spectrum trace data of the Occupied Bandwidth measurement.

Conditions

Measurement views: Occupied Bandwidth

Group

Fetch commands

Syntax

FETCh:OBWidth:SPECtrum?

Arguments

None

Returns

#<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the amplitude in dBm at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2. The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples

FETCH:OBWIDTH:SPECTRUM? might return #43204xxxx... (3204-byte data) for the spectrum trace data of the Occupied Bandwidth measurement.

FETCh:OBWidth:XDBBandwidth:BANDwidth? (Query Only)

Returns the x dB bandwidth in the Occupied Bandwidth measurement.

Conditions

Measurement views: Occupied Bandwidth

Group

Fetch commands

Syntax

FETCh:OBWidth:XDBBandwidth:BANDwidth?

Arguments

None

Returns

<xdBBW>::<NRf> is the x dB bandwidth in Hz.

Examples

FETCH:OBWIDTH:XDBBANDWIDTH:BANDWIDTH? might return 2.0E+6, indicating the x dB bandwidth is 2 MHz.

FETCh:OBWidth:XDBBandwidth:LEFT:FREQuency? (Query Only)

Returns the left (lower) frequency of the x dB bandwidth.

Conditions

Measurement views: Occupied Bandwidth

Group

Fetch commands

Syntax

FETCh:OBWidth:XDBBandwidth:LEFT:FREQuency?

Related Commands

FETCh:OBWidth:XDBBandwidth:RIGHt:FREQuency? (on page 1230)

Arguments

None

Returns

<xdBBW_left_freq>::<NRf> is the left frequency in Hz.

Examples

`FETCH:OBWIDTH:XDBBANDWIDTH:LEFT:FREQUENCY?` might return `1.498E+9`, indicating the left frequency is 1.498 GHz.

FETCh:OBWidth:XDBBandwidth:LEFT:LEVel? (Query Only)

Returns the level at the left frequency of the x dB bandwidth.

Conditions

Measurement views: Occupied Bandwidth

Group

Fetch commands

Syntax

`FETCh:OBWidth:XDBBandwidth:LEFT:LEVel?`

Related Commands

FETCh:OBWidth:XDBBandwidth:RIGHt:LEVel? (on page 1230)

Arguments

None

Returns

`<xdBBW_left_level>::<NRf>` is the level at the left frequency in dB.

Examples

`FETCH:OBWIDTH:XDBBANDWIDTH:LEFT:LEVEL?` might return `-23.5`, indicating the level at the left frequency is -23.5 dB.

FETCh:OBWidth:XDBBandwidth:POWer? (Query Only)

Returns the reference power in the x dB bandwidth measurement.

Conditions

Measurement views: Occupied Bandwidth

Group

Fetch commands

Syntax

`FETCh:OBWidth:XDBBandwidth:POWer?`

Arguments

None

Returns

`<xdBBW_ref_power>::<NRf>` is the reference power in dBm. The unit can be changed by the `[SENSe]:POWer:UNITs` command.

Examples

`FETCH:OBWIDTH:XDBBANDWIDTH:POWER?` might return `-10.0`, indicating the reference power is -10 dBm.

FETCh:OBWidth:XDBBandwidth:RIGHt:FREQuency? (Query Only)

Returns the right (higher) frequency of the x dB bandwidth.

Conditions

Measurement views: Occupied Bandwidth

Group

Fetch commands

Syntax

`FETCh:OBWidth:XDBBandwidth:RIGHt:FREQuency?`

Related Commands

FETCh:OBWidth:XDBBandwidth:LEFT:FREQuency? (on page 1228)

Arguments

None

Returns

`<xdBBW_right_freq>::<NRf>` is the right frequency in Hz.

Examples

`FETCH:OBWIDTH:XDBBANDWIDTH:RIGHT:FREQUENCY?` might return `1.502E+9`, indicating the right frequency is 1.502 GHz.

FETCh:OBWidth:XDBBandwidth:RIGHt:LEVel? (Query Only)

Returns the level at the right frequency of the x dB bandwidth.

Conditions

Measurement views: Occupied Bandwidth

Group

Fetch commands

Syntax

`FETCh:OBWidth:XDBBandwidth:RIGHt:LEVel?`

Related Commands

FETCh:OBWidth:XDBBandwidth:LEFT:LEVel? (on page 1229)

Arguments

None

Returns

<xdBBW_right_level>::<NRf> is the level at the right frequency in dB.

Examples

FETCH:OBWIDTH:XDBBANDWIDTH:RIGHT:LEVEL? might return -23.5, indicating the level at the right frequency is -23.5 dB.

FETCh:OFDM:APOWer? (Query Only)

Returns the average power in the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax

FETCh:OFDM:APOWer?

Arguments

None

Returns

<apower>::<NRf> the average power in dB.

Examples

FETCH:OFDM:APOWER? might return -23.4584459235, indicating the average power is –23.46 dB.

FETCh:OFDM:APOWer:PEAK? (Query Only)

Returns the peak-to-average power in the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax

FETCh:OFDM:APOWer:PEAK?

Arguments

None

Returns

<apower_peak>::<NRf> is the peak-to-average power in dBm.

Examples

FETCH:OFDM:APOWER:PEAK? might return 10.4140096289, indicating that peak-to-average power is 10.41 dBm.

FETCh:OFDM:CONSte:MAGNitude? (Query Only)

Returns the constellation magnitude data for the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax

`FETCh:OFDM:CONSte:MAGNitude?`

Related Commands

FETCh:OFDM:CONSte:PHASe? (on page 1232)

FETCh:OFDM:CONSte:TYPE? (on page 1233)

FETCh:OFDM:CONSte:VALue? (on page 1234)

Arguments

None

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`FETCH:OFDM:CONSTE:MAGNITUDE?` might return `#510400xxxx...` (10400-byte data) for the measurement.

FETCh:OFDM:CONSte:PHASe? (Query Only)

Returns the constellation phase values of the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax

`FETCh:OFDM:CONSte:PHASe?`

Related Commands

FETCh:OFDM:CONSte:MAGNitude? (on page 1232)

FETCh:OFDM:CONSte:TYPE? (on page 1233)

FETCh:OFDM:CONSte:VALue? (on page 1234)

Arguments

None

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the nth data point in 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`FETCH:OFDM:CONSTE:PHASE?` might return `#510400xxxx...` (10400-byte data) for the measurement.

FETCh:OFDM:CONSte:TYPE? (Query Only)

Returns the constellation context value of the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax

`FETCh:OFDM:CONSte:TYPE?`

Related Commands

FETCh:OFDM:CONSte:MAGNitude? (on page 1232)

FETCh:OFDM:CONSte:PHASe? (on page 1232)

FETCh:OFDM:CONSte:VALue? (on page 1234)

Arguments

None

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the nth data point in 16-bit little endian integers. The integers must be translated to the proper context as listed below.

Integer value	Type (context)
0	Pilot

Integer value	Type (context)
1	Data
2	Unused or Null

Examples

`FETCH:OFDM:CONSTE:TYPE?` might return `#41352` (1352-byte data) for the measurement. The actual data must be decoded to the context type using the table above.

FETCh:OFDM:CONSte:VALue? (Query Only)

Returns the constellation value (decoded symbol) of the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax

`FETCh:OFDM:CONSte:VALue?`

Related Commands

FETCh:OFDM:CONSte:MAGNitude? (on page 1232)

FETCh:OFDM:CONSte:PHASe? (on page 1232)

FETCh:OFDM:CONSte:TYPE? (on page 1233)

Arguments

None

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the nth data point in 16-bit little endian integers.

Examples

`FETCH:OFDM:CONSTE:VALUE?` might return `#41352` indicating 1352 bytes of data for the measurement.

FETCh:OFDM:CPE? (Query Only)

Returns the Common Pilot Error magnitude for the OFDM measurement

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax

`FETCh:OFDM:CPE?`

Arguments

None

Returns

`<NRf>` the RMS magnitude error in percent.

Examples

`FETCH:OFDM:CPE?` might return `3.7868041505`, indicating that the CPE is 3.787%.

FETCh:OFDM:CRESponse:MAGNitude? (Query Only)

Returns the channel response magnitude data for the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax

`FETCh:OFDM:CRESponse:MAGNitude?`

Arguments

None

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`FETCH:OFDM:CRESPONSE:MAGNITUDE?` might return `#3804xxxx. . .` (804 byte data) for the measurement.

FETCh:OFDM:CRESponse:PHASe? (Query Only)

Returns the channel response phase data for the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax

`FETCh:OFDM:CRESponse:PHASe?`

Arguments

None

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`FETCH:OFDM:CRESPONSE:PHASE?` might return `#3804xxxx. . .` (804 byte data) for the measurement.

FETCh:OFDM:EVM:PEAK:DECibel:ALL? (Query Only)

Returns the peak EVM data for all subcarriers in the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax

`FETCh:OFDM:EVM:PEAK:DECibel:ALL?`

Arguments

None

Returns

`<NRf>` data in dB.

Examples

`FETCH:OFDM:EVM:PEAK:DECIBEL:ALL?` might return `-20.1872549032`, indicating the data is –20.19 dB for the measurement.

FETCh:OFDM:EVM:PEAK:DECibel:DATA? (Query Only)

Returns the peak EVM data for the data subcarriers in the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax

`FETCh:OFDM:EVM:PEAK:DECibel:DATA?`

Arguments

None

Returns

`<NRf>` data in dB.

Examples

`FETCH:OFDM:EVM:PEAK:DECIBEL:DATA?` might return `-20.1872549032` indicating the data is –20.19 dB for the measurement.

FETCh:OFDM:EVM:PEAK:DECibel:PILots? (Query Only)

Returns the peak EVM data for the pilot subcarriers in the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax

`FETCh:OFDM:EVM:PEAK:DECibel:PILots?`

Arguments

None

Returns

`<NRf>` data in dB.

Examples

`FETCH:OFDM:EVM:PEAK:DECIBEL:PILOTS?` might return `-20.2872549032`, indicating the data is –20.29 dB for the measurement.

FETCh:OFDM:EVM:PEAK:PERCent:ALL? (Query Only)

Returns the peak EVM data for all subcarriers in the OFDM measurement as a percent.

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax
`FETCh:OFDM:EVM:PEAK:PERCent:ALL?`

Arguments
None

Returns
`<NRf>` data in percent.

Examples
`FETCH:OFDM:EVM:PEAK:PERCENT:ALL?` might return `19.3223863840`, indicating the data is 19.322% for the measurement.

FETCh:OFDM:EVM:PEAK:PERCent:DATA? (Query Only)
Returns the peak EVM data for the data subcarriers in the OFDM measurement as a percent.

Conditions
Measurement views: OFDM

Group
Fetch commands

Syntax
`FETCh:OFDM:EVM:PEAK:PERCent:DATA?`

Arguments
None

Returns
`<NRf>` data in percent.

Examples
`FETCH:OFDM:EVM:PEAK:PERCENT:DATA?` might return `19.3223863840`, indicating the data is 19.322% for the measurement.

FETCh:OFDM:EVM:PEAK:PERCent:PILots? (Query Only)
Returns the peak EVM data for the pilot subcarriers in the OFDM measurement as a percent.

Conditions
Measurement views: OFDM

Group
Fetch commands

Syntax
`FETCh:OFDM:EVM:PEAK:PERCent:PILots?`

Arguments

None

Returns

<NRf> data in percent.

Examples

FETCH:OFDM:EVM:PEAK:PERCENT:PILOTS? might return 10.8543863840, indicating the data is 10.854% for the measurement.

FETCh:OFDM:EVM:PEAK:SCARrier:ALL? (Query Only)

Returns the peak EVM data for all subcarriers at the subcarrier level in the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax

FETCh:OFDM:EVM:PEAK:SCARrier:ALL?

Arguments

None

Returns

<NR1>

Examples

FETCH:OFDM:EVM:PEAK:SCARRIER:ALL? might return 99.00000, indicating the data is 99 for the measurement.

FETCh:OFDM:EVM:PEAK:SCARrier:DATA? (Query Only)

Returns the peak EVM data for all data subcarriers at the subcarrier level in the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax

FETCh:OFDM:EVM:PEAK:SCARrier:DATA?

Arguments

None

Returns

<NR1>

Examples

`FETCH:OFDM:EVM:PEAK:SCARRIER:DATA?` might return `99.00000`, indicating the data is 99 for the measurement.

FETCh:OFDM:EVM:PEAK:SCARrier:PILots? (Query Only)

Returns the peak EVM data for the pilot subcarriers at the subxcarrier level in the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax

`FETCh:OFDM:EVM:PEAK:SCARrier:PILots?`

Arguments

None

Returns

<NR1>

Examples

`FETCH:OFDM:EVM:PEAK:SCARRIER:PILOTS?` might return `88.00000`, indicating the data is 88 for the measurement

FETCh:OFDM:EVM:PEAK:SYMBol:ALL? (Query Only)

Returns the peak EVM data for all subcarriers at the symbol level in the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax

`FETCh:OFDM:EVM:PEAK:SYMBol:ALL?`

Arguments

None

Returns

<NR1>

Examples

`FETCH:OFDM:EVM:PEAK:SYMBOL:ALL?` might return 2.00000 indicating the data is 2 for the measurement.

FETCh:OFDM:EVM:PEAK:SYMBol:DATA? (Query Only)

Returns the peak EVM data for the data subcarriers at the symbol level in the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax

`FETCh:OFDM:EVM:PEAK:SYMBol:DATA?`

Arguments

None

Returns

<NR1>

Examples

`FETCH:OFDM:EVM:PEAK:SYMBOL:DATA?` might return `2.00000`, indicating the data is 2 for the measurement.

FETCh:OFDM:EVM:PEAK:SYMBol:PILots? (Query Only)

Returns the peak EVM data for the pilot subcarriers at the symbol level in the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax

`FETCh:OFDM:EVM:PEAK:SYMBol:PILots?`

Arguments

None

Returns

<NR1>

Examples

`FETCH:OFDM:EVM:PEAK:SYMBOL:PILOTS?` might return `1.00000`, indicating the data is `1` for the measurement.

FETCh:OFDM:EVM:RMS:DECibel:ALL? (Query Only)

Returns the RMS EVM data for all subcarriers in the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax
`FETCh:OFDM:EVM:RMS:DECibel:ALL?`

Arguments
None

Returns
<NRf> data in dB.

Examples
`FETCH:OFDM:EVM:RMS:DECIBEL:ALL?` might return `-26.9012093267`, indicating the data is –26.90 for the measurement.

FETCh:OFDM:EVM:RMS:DECibel:DATA? (Query Only)
Returns the RMS EVM data for the data subcarriers in the OFDM measurement.

Conditions
Measurement views: OFDM

Group
Fetch commands

Syntax
`FETCh:OFDM:EVM:RMS:DECibel:DATA?`

Arguments
None

Returns
<NRf> data in dB.

Examples
`FETCH:OFDM:EVM:RMS:DECIBEL:DATA?` might return `-26.8477116269`, indicating the data is –26.85 for the measurement.

FETCh:OFDM:EVM:RMS:DECibel:PILots? (Query Only)
Returns the RMS EVM data for the pilot subcarriers in the OFDM measurement.

Conditions
Measurement views: OFDM

Group
Fetch commands

Syntax
`FETCh:OFDM:EVM:RMS:DECibel:PILots?`

Arguments

None

Returns

<NRf> data in dB.

Examples

`FETCH:OFDM:EVM:RMS:DECIBEL:PILOTS?` might return `-33.0589143032`, indicating the data is –33.06 dB.

FETCh:OFDM:EVM:RMS:PERCent:ALL? (Query Only)

Returns the peak RMS data for all subcarriers in the OFDM measurement as a percent.

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax

`FETCh:OFDM:EVM:RMS:PERCent:ALL?`

Arguments

None

Returns

<NRf> data in percent.

Examples

`FETCH:OFDM:EVM:RMS:PERCENT:ALL?` might return `225.0743627548`, indicating the data is 225.074% for the measurement.

FETCh:OFDM:EVM:RMS:PERCent:DATA? (Query Only)

Returns the peak RMS data for the data subcarriers in the OFDM measurement as a percent.

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax

`FETCh:OFDM:EVM:RMS:PERCent:DATA?`

Arguments

None

Returns

<NRf> data in percent.

Examples

FETCH:OFDM:EVM:RMS:PERCENT:DATA? might return 260.6973409653, indicating that the data is 260.697% for the measurement.

FETCh:OFDM:EVM:RMS:PERCent:PILots? (Query Only)

Returns the peak RMS data for the pilot subcarriers in the OFDM measurement as a percent.

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax

FETCh:OFDM:EVM:RMS:PERCent:PILots?

Arguments

None

Returns

<NRf> data in percent.

Examples

FETCH:OFDM:EVM:RMS:PERCENT:PILOTS? might return 210.8103863840, indicating the data is 210.810% for the measurement.

FETCh:OFDM:EVM:TRACe<x>? (Query Only)

Returns the EVM trace data for the OFDM measurement.

When <x> is 1, the parameter is Matrix.

When <x> is 2, the parameter is Average versus Symbols.

When <x> is 3, the parameter is Average versus Subcarrier.

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax

FETCh:OFDM:EVM:TRACe<x>?

Arguments

None

Returns

#<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

FETCH:OFDM:EVM:TRACE2? might return #252xxxx. . . (52 byte data) for the Average versus Symbols measurement.

FETCh:OFDM:FERRor? (Query Only)

Returns the Frequency error reading for the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax

FETCh:OFDM:FERRor?

Arguments

None

Returns

<NRf> frequency error.

Examples

FETCH:OFDM:FERROR? might return 82.8617142098E+3, indicating the frequency error was 82.86 kHz.

FETCh:OFDM:FLATness:PASS? (Query Only)

Queries whether the average power levels of subcarriers across the signal bandwidth on the OFDM Spectral Flatness display remain within the limits defined for a particular standard.

Conditions

Measurement view: OFDM

This command requires Option 22, "OFDM Measurements".

Group

Fetch commands

Syntax

FETCh:OFDM:FLATness:PASS?

Returns

Returns "1" in the case that all data remains within the defined limits, and "0" otherwise.

Examples

`FETCH:OFDM:FLAT:PASS?` might return 1, which means that the average power levels of subcarriers across the signal bandwidth remain within the defined limits.

FETCh:OFDM:FLATness:RESult? (Query Only)

Returns the summary results of each segment of the OFDM Spectral Flatness display.

Conditions

Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group

Fetch commands

Syntax

`FETCh:OFDM:FLATness:RESult?`

Returns

The data is returned as 12 comma delineated text values. The specific values are :

- minimum subcarrier within segment 1
- minimum subcarrier average energy deviation from the global average within segment 1
- deviation the of minimum subcarrier average from the allowed value in segment 1
- maximum subcarrier within segment 1
- maximum subcarrier average energy deviation from the global average within segment 1
- deviation of the maximum subcarrier average from the allowed value in segment 1
- minimum subcarrier within segment 2
- minimum subcarrier average energy deviation from the global average within segment 2
- deviation the of minimum subcarrier average from the allowed value in segment 2
- maximum subcarrier within segment 2
- maximum subcarrier average energy deviation from the global average within segment 2
- deviation of the maximum subcarrier average from the allowed value in segment 2

Examples

`FETCH:OFDM:FLAT:RES?` might return `-6.0000000000,-17.0626174659E-3,3.9829373825,12.0000000000,19.979260861` which represents the summary results of each segment of the OFDM Spectral Flatness display.

FETCh:OFDM:FLATness:TRACe<x>? (Query Only)

Returns the OFDM Spectral Flatness trace data. When <x> is 1, the parameter is Matrix (symbol deviation per subcarrier). When<x> is 2, the parameter is Average Deviation vs Subcarrier.

Conditions

Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group

Fetch commands

Syntax

`FETCh:OFDM:FLATness:TRACe<x>?`

Returns

`#<num_digit><num_byte><data(1)><data(2)>..<data(n)>`

Where:

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`FETCH:OFDM:FLAT:TRACE2?` might return `#510400..` (10400-byte data) which represents the data for the Average Deviation vs. Subcarrier trace.

FETCh:OFDM:GAIN:IMBalance? (Query Only)

Returns the gain imbalance for the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax

`FETCh:OFDM:GAIN:IMBalance?`

Arguments

None

Returns

<NRf> gain imbalance in dB.

Examples

`FETCH:OFDM:GAIN:IMBALANCE?` might return `-57.746E-3`, indicating that the gain imbalance is −0.057746 dB.

FETCh:OFDM:IQ:ORIGin:OFFSet? (Query Only)

Returns the IQ origin offset for the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax

FETCh:OFDM:IQ:ORIGin:OFFSet?

Arguments

None

Returns

<NRf> gain imbalance in dB.

Examples

FETCH:OFDM:IQ:ORIGIN:OFFSET? might return −53.47017 indicating that the IQ origin offset is −53.47 dB.

FETCh:OFDM:MERRor:TRACe<x>? (Query Only)

Returns the magnitude error trace data for the OFDM measurement.

When <x> is 1, the parameter is Matrix.

When <x> is 2, the parameter is Average versus Symbols.

When <x> is 3, the parameter is Average versus Subcarrier.

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax

FETCh:OFDM:MERRor:TRACe<x>?

Arguments

None

Returns

#<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

FETCH:OFDM:MERROR:TRACE2 might return #3108xxxx. . . (108-byte data) for the average versus symbols trace measurement.

FETCh:OFDM:PACKet:DIRection? (Query Only)

Returns the direction of the packet in the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax

`FETCh:OFDM:PACKet:DIRection?`

Arguments

None

Returns

One of the following values.

`UP` specifies that the packet is in the up direction.

`DOWN` specifies that the packet is in the down direction.

`BOTH` specifies that the packet is both directions.

Examples

`FETCH:OFDM:PACKET:DIRECTION?` might return `DOWN`, indicating that the packet was in the down direction.

FETCh:OFDM:PERRor:TRACe<x>? (Query Only)

Returns the phase error trace data for the OFDM measurement.

When <x> is 1, the parameter is Matrix.

When <x> is 2, the parameter is Average versus Symbols.

When <x> is 3, the parameter is Average versus Subcarrier.

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax

`FETCh:OFDM:PERRor:TRACe<x>?`

Arguments

None

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

FETCH:OFDM:PERROR:TRACE2 might return #3100xxxx. . . (100-byte data) for the average versus symbols trace measurement.

FETCh:OFDM:POWer:TRACe<x>? (Query Only)

Returns the power trace data for the OFDM measurement. When <x> is 1, the parameter is Matrix.

When <x> is 2, the parameter is Average versus Symbols. When <x> is 3, the parameter is Average versus Subcarrier.

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax

FETCh:OFDM:POWer:TRACe<x>?

Arguments

None

Returns

#<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

FETCH:OFDM:POWER:TRACE2 might return #296xxxx. . . (96-byte data) for the average versus symbols trace measurement.

FETCh:OFDM:QUADrature:OFFSet? (Query Only)

Returns the quadrature offset in the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax

FETCh:OFDM:QUADrature:OFFSet?

Arguments

None

Returns

<NRf> quadrature offset in degrees.

Examples

`FETCH:OFDM:QUADRATURE:OFFSET?` might return `-99.9`, indicating that the offset is -99°.

FETCh:OFDM:SCARriers? (Query Only)

Returns the number of subcarriers in the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax

`FETCh:OFDM:SCARriers?`

Arguments

None

Returns

<NR1> number of subcarriers.

Examples

`FETCH:OFDM:SCARRIERS?` might return `200.0000`, indicating that there are 200 subcarriers in the measurement.

FETCh:OFDM:SCARriers:SPACing? (Query Only)

Returns the subcarrier spacing for the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax

`FETCh:OFDM:SCARriers:SPACing?`

Arguments

None

Returns

<NRf> subcarrier spacing.

Examples

`FETCH:OFDM:SCARRIERS:SPACING?` might return `90.000E+3`, indicating that the subcarrier spacing is 90.000 kHz.

FETCh:OFDM:STABle:VALUe? (Query Only)

Returns the number of bytes in the symbol table for the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax

`FETCh:OFDM:STABle:VALUe?`

Arguments

None

Returns

`#<num_digit><num_byte>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

Examples

`FETCH:OFDM:STABLE:VALUE?` might return `#43848` indicating that there are 3848 bytes in the symbol table.

FETCh:OFDM:SYMBol:CERRor? (Query Only)

Returns the symbol clock error for the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax

`FETCh:OFDM:SYMBol:CERRor?`

Arguments

None

Returns

<NRf> symbol clock error in ppm.

Examples

`FETCH:OFDM:SYMBOL:CERROR?` might return `422.7135479929`, indicating that the symbol clock error was 422.714 ppm.

FETCh:OFDM:SYMBol:COUNt? (Query Only)

Returns the number of symbols for the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Fetch commands

Syntax

`FETCh:OFDM:SYMBol:COUNt?`

Arguments

None

Returns

<NR1> number of symbols in the measurement.

Examples

`FETCH:OFDM:SYMBOL:COUNT?` might return `25.00000`, indicating that there were 25 symbols for the measurement.

FETCh:P25:CONSte:FERRor? (Query Only)

Returns the frequency error in Hz. The frequency error is the difference between the measured carrier frequency of the signal and the user-selected center frequency of the analyzer.

When Frequency Error is in Auto mode, this query returns the frequency error value. When in Manual mode, it returns the frequency offset.

Conditions

Measurement view: P25 Constellation

Group

Fetch commands

Syntax

FETCh:P25:CONSte:FERRor?

Returns

<freq_error>::<NRf>, which is the frequency error in Hz.

Examples

`FETCH:P25:CONSTE:FERROR?` might return `-10.7E+3`, which is a frequency error of -10.7 kHz.

FETCh:P25:CONSte:TRACe? (Query Only)

Returns the P25 Constellation display trace data.

Conditions

Measurement view: P25 Constellation

Group

Fetch commands

Syntax

`FETCh:P25:CONSte:TRACe?`

Returns

`#<num_digit><num_byte><I(1)><Q(1)><I(2)><Q(2)>...<I(n)><Q(n)>`

Where:

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<I(n)>` and `<Q(n)>` are the normalized I- and Q- coordinate values at the *n*th data point. 4-byte little endian floating point format specified in IEEE 488.2.

Examples

`FETCH:P25:CONSTE:TRACE?` might return `#43848xxxx...` (3848-byte data) for the P25 Constellation trace data.

FETCh:P25:EDIagram:FDEViation? (Query Only)

Returns the Frequency Deviation vs Time trace data with the X value.

Conditions

Measurement view: P25 Eye Diagram

Group

Fetch commands

Syntax

`FETCh:P25:EDIagram:FDEViation?`

Returns

`#<num_digit><num_byte><Y(1)><X(1)><Y(2)><X(2)>...<Y(n)><X(n)>`

Where:

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<Y(n)>` and `<X(n)>` are the frequency deviation in Hz and time (symbols) coordinate pair at the *n*th data point. 4-byte little endian floating point format specified in IEEE 488.2.

Examples

`FETCH:P25:EDIAGRAM:FDEVIATION?` might return `#3160xxxx...` (160-byte data) for the Frequency Deviation vs Time trace.

FETCh:P25:EDIagram:FERRor? (Query Only)

Returns the frequency error in the P25 Eye Diagram display.

When Frequency Error is in Auto mode, this query returns the frequency error value. When in Manual mode, it returns the frequency offset.

Conditions

Measurement view: P25 Eye Diagram

Group

Fetch commands

Syntax

`FETCh:P25:EDIagram:FERRor?`

Returns

<feq_error>::<NRf> is the frequency error in Hz.

Examples

`FETCH:P25:EDIAGRAM:FERROR?` might return `-10.7E+3`, indicating the frequency Error is -10.7 kHz.

FETCh:P25:EDIagram:I? (Query Only)

Returns the I versus Time trace data in the P25 Eye Diagram display.

Conditions

Measurement view: P25 Eye Diagram

Group

Fetch commands

Syntax

`FETCh:P25:EDIagram:I?`

Returns

`#<num_digit><num_byte><Y(1)><X(1)><Y(2)><X(2)>...<Y(n)><X(n)>`

Where:

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<Y(n)>` and `<X(n)>` are the normalized I- and time (symbols) coordinate values at the n^{th} data point. 4-byte little endian floating point format specified in IEEE 488.2.

Examples

`FETCH:P25:EDIAGRAM:I?` might return `#3160xxxx...` (160-byte data) for the I versus Time trace.

FETCh:P25:EDIagram:Q? (Query Only)

Returns the Q versus Time trace data in the P25 Eye Diagram display.

Conditions

Measurement view: P25 Eye Diagram

Group

Fetch commands

Syntax

`FETCh:P25:EDIagram:Q?`

Returns

`#<num_digit><num_byte><Y(1)><X(1)><Y(2)><X(2)>...<Y(n)><X(n)>`

Where:

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<Y(n)>` and `<X(n)>` are the normalized Q- and time (symbols) coordinate values at the *n*th data point. 4-byte little endian floating point format specified in IEEE 488.2.

Examples

`FETCH:P25:EDIAGRAM:Q?` might return `#3160xxxx...` (160-byte data) for the Q Versus Time trace.

FETCh:P25:PVTime:BURSt:WIDTh? (Query Only)

Returns the value of the measured time width of the burst packet from poweron to powerdown (or the end of waveform, if that occurs before powerdown), in seconds, for the P25 Power vs. Time trace.

Conditions

Measurement view: P25 Power vs. Time

Group

Fetch commands

Syntax

`FETCh:P25:PVTime:BURSt:WIDTh?`

Returns

Floating point number that represents the value of the measured time width of the burst packet from power on to power down (or the end of waveform, if that occurs before power down), in seconds.

Examples

`FETCH:P25:PVTIME:BURST:WIDTH?` might return `156.0000000000E-6`, which represents a width of the burst packet of 156 µs.

FETCh:P25:PVTime:FALL:TIME:STARt? (Query Only)

Returns the value of the start time of the falling edge, in seconds.

Conditions

Measurement view: P25 Power vs. Time

Group

Fetch commands

Syntax

```
FETCh:P25:PVTime:FALL:TIME:STARt?
```

Related Commands

FETCh:P25:PVTime:FALL:TIME:STOP? (on page 1257)

Returns

Floating point number which represents the value of the start time of the falling edge of the burst, in seconds.

Examples

`FETCH:P25:PVTIME:FALL:TIME:START?` might return `6e-6`, which indicates that the start time of the falling edge is at 6 µs.

FETCh:P25:PVTime:FALL:TIME:STOP? (Query Only)

Returns the value of the stop time of the falling edge, in seconds.

Conditions

Measurement view: P25 Power vs. Time

Group

Fetch commands

Syntax

```
FETCh:P25:PVTime:FALL:TIME:STOP?
```

Related Commands

FETCh:P25:PVTime:FALL:TIME:STARt? (on page 1256)

Returns

Floating point number which represents the value of the stop time of the falling edge of the burst, in seconds.

Examples

`FETCH:P25:PVTIME:FALL:TIME:STOP?` might return `7e-6`, which indicates the stop time of the falling edge is at 7 µs.

FETCh:P25:PVTime:RISE:TIME? (Query Only)

Returns the value of the width of the rising edge, in seconds.

Conditions

Measurement view: P25 Power vs. Time

Group

Fetch commands

Syntax

```
FETCh:P25:PVTime:RISE:TIME?
```

Related Commands

FETCh:P25:PVTime:RTIMe? (on page 1259)

Returns

Floating point number which represents the value of the start time of the rising edge of the burst, in seconds.

Examples

`FETCH:P25:PVTIME:RISE:TIME?` might return `6e-6`, which indicates the start time of the rising edge is 6 µs.

FETCh:P25:PVTime:RISE:TIME:STARt? (Query Only)

Returns the value of the start time of the rising edge, in seconds.

Conditions

Measurement view: P25 Power vs. Time

Group

Fetch commands

Syntax

`FETCh:P25:PVTime:RISE:TIME:STARt?`

Related Commands

FETCh:P25:PVTime:RISE:TIME? (on page 1257)

FETCh:P25:PVTime:RTIMe? (on page 1259)

Returns

Floating point number which represents the value of the start time of the rising edge of the burst, in seconds.

Examples

`FETCH:P25:PVTIME:RISE:TIME:START?` might return `6e-6`, which indicates the start time of the rising edge is at 6 µs.

FETCh:P25:PVTime:RISE:TIME:STOP? (Query Only)

Returns the value of the stop time of the rising edge, in seconds.

Conditions

Measurement view: P25 Power vs. Time

Group

Fetch commands

Syntax

`FETCh:P25:PVTime:RISE:TIME:STOP?`

Related Commands

FETCh:P25:PVTime:RISE:TIME:STARt? (on page 1257)

FETCh:P25:PVTime:RTIMe? (on page 1259)

FETCh:P25:PVTime:RISE:TIME? (on page 1257)

Returns

Floating point number which represents the value of the stop time of the rising edge of the burst, in seconds.

Examples

`FETCH:P25:PVTIME:RISE:TIME:STOP?` might return `7e-6`, which indicates the width of the rising edge is 7 µs.

FETCh:P25:PVTime:RTIMe? (Query Only)

Returns the value of the width of the rising edge, in seconds.

Conditions

Measurement view: P25 Power vs. Time

Group

Fetch commands

Syntax

`FETCh:P25:PVTime:RTIMe?`

Related Commands

FETCh:P25:PVTime:RISE:TIME? (on page 1257)

Returns

Floating point number which represents the value of the width of the rising edge of the burst, in seconds.

Examples

`FETCH:P25:PVTIME:RTIME?` might return `7e-6`, which indicates the width of the rising edge is 7 µs.

FETCh:P25:PVTime:TRACe[:Y]? (Query Only)

Returns the vertical values (power) for the P25 Power vs. Time trace.

Conditions

Measurement view: P25 Power vs. Time

Group

Fetch commands

Syntax

`FETCh:P25:PVTime:TRACe[:Y]?`

Returns

`#<num_digit><num_byte><data(1)><data(2)>..<data(n)>`

Where:

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

FETCH:P25:PVTIME:TRACE[:Y]? might return #43204 (3204-byte data), which represents the vertical values (power) for the P25 Power vs. Time trace.

FETCh:P25:SUMMary:LIMIt: SRA (Query Only)

Returns Limit of Symbol Rate Accuracy measurement results.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Fetch commands

Syntax

FETCh:P25:SUMMary:LIMIt:SRA

Arguments

None

Returns

might return PASS, FAIL, or NA.

Examples

FETCH:P25:SUMMARY:LIMIT:SRA

FETCh:P25:SUMMary:MODUlation:MEASurement:FIDelity (Query Only)

Returns the Modulation Fidelity measurement result.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Fetch commands

Syntax

FETCh:P25:SUMMary:MODUlation:MEASurement:FIDelity

Arguments

None

Examples

FETCH:P25:SUMMARY:MODULATION:MEASUREMENT:FIDELITY

FETCh:P25:SUMMary:MODUlation:MEASurement:FIDelity:STATus (Query Only)

Returns the Modulation Fidelity measurement status.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Fetch commands

Syntax

`FETCh:P25:SUMMary:MODUlation:MEASurement:FIDelity:STATus`

Arguments

None

Returns

PASS means the result passed.

FAIL means the result failed.

NA means there is nothing to return.

Examples

`FETCH:P25:SUMMARY:MODULATION:MEASUREMENT:FIDELITY:STATUS` might return `PASS`, `FAIL`, or `NA`.

FETCH:P25:SUMMary:MODUlation:MEASurement:FREQuency:.DEViation:POSPeak

Returns the Frequency Deviation, positive peak measurement results.

Conditions

Measurement views: P25 Summary

Group

Fetch commands

Syntax

`FETCH:P25:SUMMary:MODUlation:MEASurement:FREQuency:.DEVIation:POSPeak`

Arguments

<integer>,<integer>

Returns

Frequency Deviation, positive peak, result

Examples

FETCH:P25:SUMMARY:MODULATION:MEASUREMENT:FREQUENCY:.DEVIATION:POSPEAK <-2,-6> indicates the coordinate value of the Frequency Deviation positive peak.

FETCh:P25:SUMMary:MODUlation:MEASurement:FREQuency:.DEVIation:POSPeak:STA

Returns the status of Frequency Deviation, positive peak status.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Fetch commands

Syntax

FETCh:P25:SUMMary:MODUlation:MEASurement:FREQuency:.DEVIation:POSPeak:STATus

Arguments

None

Examples

FETCH:P25:SUMMARY:MODULATION:MEASUREMENT:FREQUENCY:.DEVIATION:POSPEAK:STATUS might return PASS, FAIL or NA.

FETCH:P25:SUMMary:MODUlation:MEASurement:FREQuency:ACCUracy (Query Only)

Returns the Operational Frequency Accuracy measurement results.

Conditions

Measurement views: P25 Summary

Group

Fetch commands

Syntax

FETCH:P25:SUMMary:MODUlation:MEASurement:FREQuency:ACCUracy

Arguments

None

Returns

Frequency Accuracy measurement result

Examples

FETCH:P25:SUMMARY:MODULATION:MEASUREMENT:FREQUENCY:ACCURACY

FETCH:P25:SUMMary:MODUlation:MEASurement:FREQuency:ACCUracy (Query Only)

Returns the Operational Frequency Accuracy measurement results in the P25 Summary display.

Conditions

Measurement view: P25 Summary

Group

Fetch commands

Syntax

`FETCH:P25:SUMMary:MODUlation:MEASurement:FREQuency:ACCUracy?`

Arguments

None

Returns

<NRf> Returns Operational Frequency Accuracy

Examples

`FETCH:P25:SUMMARY:MODULATION:MEASUREMENT:FREQUENCY:ACCURACY?` might return `25.62`, indicating the Operational Frequency Accuracy is 25.62 ppm.

FETCh:P25:SUMMary:MODUlation:MEASurement:FREQuency:ACCUracy:STATus (Query Only)

Returns the status of Frequency Accuracy measurement's result.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Fetch commands

Syntax

`FETCh:P25:SUMMary:MODUlation:MEASurement:FREQuency:ACCUracy:STATus`

Arguments

None

Examples

`FETCH:P25:SUMMARY:MODULATION:MEASUREMENT:FREQUENCY:ACCURACY:STATUS` might return `PASS`, `FAIL`, or `NA`.

FETCh:P25:SUMMary:MODUlation:MEASurement:FREQuency:DEViation:NEGPeak:STATus (Query Only)

Queries the status of Frequency Deviation, negative peak result in the P25 Summary display.

Conditions

Measurement view: P25 Summary

Group

Fetch commands

Syntax

`FETCh:P25:SUMMary:MODUlation:MEASurement:FREQuency:DEVIation:NEGPeak:STATus?`

Arguments

None

Returns

<String>

PASS indicates that the test passed.

FAIL indicates that the test failed.

NA indicates that the test is not Applicable.

Examples

`FETCH:P25:SUMMARY:MODULATION:MEASUREMENT:FREQUENCY:DEVIATION:NEGPEAK:STATUS?` might return `Pass`, indicating that the Frequency Deviation for negative peak test result is Pass.

FETCH:P25:SUMMary:MODUlation:MEASurement:FREQuency:DEViation:NEGPeak (Query Only)

Returns the Frequency Deviation, negative peak, measurement results.

Conditions

Measurement views: P25 Summary

Group

Fetch commands

Syntax

`FETCH:P25:SUMMary:MODUlation:MEASurement:FREQuency:DEVIation:NEGPeak`

Arguments

None

Returns

Frequency Deviation measurement result

Examples

FETCH:P25:SUMMARY:MODULATION:MEASUREMENT:FREQUENCY:DEVIATION:NEGPEAK

FETCh:P25:SUMMary:MODUlation:MEASurement:FREQuency:DEVIation:POSPeak:STATus (Query Only)

Queries the status of Frequency Deviation, positive peak result in the P25 Summary display.

Conditions

Measurement view: P25 Summary

Group

Fetch commands

Syntax

FETCh:P25:SUMMary:MODUlation:MEASurement:FREQuency:DEVIation:POSPeak:STATus?

Arguments

None

Returns

<String>

PASS indicates that the test passed.

FAIL indicates that the test failed.

NA indicates that the test is not Applicable.

Examples

FETCH:P25:SUMMARY:MODULATION:MEASUREMENT:FREQUENCY:DEVIATION:POSPEAK:STATUS? might return Pass, indicating that the Frequency Deviation for positive peak test result is Pass.

FETCH:P25:SUMMary:MODUlation:MEASurement:FREQuency:DEVIation:POSPeak (Query Only)

Returns the Frequency Deviation, positive peak measurement results in the P25 Summary display.

Conditions

Measurement view: P25 Summary

Group

Fetch commands

Syntax

FETCH:P25:SUMMary:MODUlation:MEASurement:FREQuency:DEVIation:POSPeak?

Arguments

None

Returns

<NRf>

Examples

`FETCH:P25:SUMMARY:MODULATION:MEASUREMENT:FREQUENCY:DEVIATION:POSPEAK?` might return `125.63`, indicating the Frequency Deviation positive peak is 125.63 Hz.

FETCH:P25:SUMMary:MODUlation:MEASurement:FREQuency:ERRor (Query Only)

Returns the Frequency Error measurement results.

Conditions

Measurement views: P25 Summary

Group

Fetch commands

Syntax

`FETCH:P25:SUMMary:MODUlation:MEASurement:FREQuency:ERRor`

Arguments

None

Returns

Frequency Error measurement result

FETCH:P25:SUMMary:MODUlation:MEASurement:SRA? (Query Only)

Returns the Symbol Rate Accuracy measurement results.

Conditions

Measurement views: P25 Summary

Group

Fetch commands

Syntax

`FETCH:P25:SUMMary:MODUlation:MEASurement:SRA?`

Arguments

None

Returns

Symbol Rate Accuracy measurement result

Examples

`FETCH:P25:SUMMARY:MODULATION:MEASUREMENT:SRA?`

FETCh:P25:SUMMary:MODUlation:MEASurement:SRA:STATus (Query Only)

Returns Status of Symbol Rate Accuracy measurement status.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Fetch commands

Syntax

`FETCh:P25:SUMMary:MODUlation:MEASurement:SRA:STATus?`

Arguments

None

Returns

PASS, FAIL or NA.

Examples

`FETCH:P25:SUMMARY:MODULATION:MEASUREMENT:SRA:STATUS?` might return `PASS`, indicating that the status is passed.

FETCH:P25:SUMMary:POWEr:MEASurement:HCPM:MAXOFF? (Query Only)

Returns the HCPM Pmax-off measurement results.

Conditions

Measurement view: P25 Summary

Group

Fetch commands

Syntax

`FETCH:P25:SUMMary:POWEr:MEASurement:HCPM:MAXOFF?`

Returns

HCPM Pmax-off measurement results.

Examples

`FETCH:P25:SUMMARY:POWER:MEASUREMENT:HCPM:MAXOFF?` would return the HCPM Pmax-off measurement result.

FETCh:P25:SUMMary:POWEr:MEASurement:HCPM:MAXOFF:STATus? (Query

Returns HCPM Pmax-off measurement status.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Fetch commands

Syntax

`FETCh:P25:SUMMary:POWEr:MEASurement:HCPM:MAXOFF:STATus?`

Arguments

None

Examples

`FETCH:P25:SUMMARY:POWER:MEASUREMENT:HCPM:MAXOFF:STATUS?` might return `PASS`, `FAIL`, or `NA`.

FETCH:P25:SUMMary:POWEr:MEASurement:HCPM:MAXON? (Query Only)

Returns the HCPM Pmax-on measurement results.

Conditions

Measurement view: P25 Summary

Group

Fetch commands

Syntax

`FETCH:P25:SUMMary:POWEr:MEASurement:HCPM:MAXON?`

Returns

HCPM Pmax-on measurement results.

Examples

`FETCH:P25:SUMMARY:POWER:MEASUREMENT:HCPM:MAXON?` would return HCPM Pmax-on measurement results.

FETCh:P25:SUMMary:POWEr:MEASurement:HCPM:MAXON:STATus? (Query Only)

Returns Status of HCPM Pmax-on measurement status.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Fetch commands

Syntax

FETCh:P25:SUMMary:POWEr:MEASurement:HCPM:MAXON:STATus?

Arguments

None

Examples

FETCH:P25:SUMMARY:POWER:MEASUREMENT:HCPM:MAXON:STATUS? might return PASS, FAIL, or NA.

FETCH:P25:SUMMary:POWEr:MEASurement:HCPM:MAXSS? (Query Only)

Returns the HCPM Pss-max measurement results.

Conditions

Measurement view: P25 Summary

Group

Fetch commands

Syntax

FETCH:P25:SUMMary:POWEr:MEASurement:HCPM:MAXSS?

Returns

HCPM Pss-max measurement results.

Examples

FETCH:P25:SUMMARY:POWER:MEASUREMENT:HCPM:MAXSS? would return the HCPM Pss-max measurement result.

FETCh:P25:SUMMary:POWEr:MEASurement:HCPM:MAXSs:STATus (Query Only)

Returns HCPM Pss-max measurement status.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Fetch commands

Syntax

FETCh:P25:SUMMary:POWEr:MEASurement:HCPM:MAXSs:STATus?

Arguments

None

Examples

FETCH:P25:SUMMARY:POWER:MEASUREMENT:HCPM:MAXSS:STATUS? might return PASS, FAIL, or NA.

FETCH:P25:SUMMary:POWEr:MEASurement:HCPM:MINImum? (Query Only)

Returns the HCPM Pss-min measurement results.

Conditions

Measurement view: P25 Summary

Group

Fetch commands

Syntax

`FETCH:P25:SUMMary:POWEr:MEASurement:HCPM:MINImum?`

Returns

HCPM Pss-min measurement results.

Examples

`FETCH:P25:SUMMARY:POWER:MEASUREMENT:HCPM:MINIMUM?` would return the HCPM Pss-min measurement result.

FETCh:P25:SUMMary:POWEr:MEASurement:HCPM:MINImum:STATus? (Query Only)

Returns status of HCPM Pss-min measurement status based on the limit.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Fetch commands

Syntax

`FETCh:P25:SUMMary:POWEr:MEASurement:HCPM:MINImum:STATus?`

Arguments

None

Examples

`FETCH:P25:SUMMARY:POWER:MEASUREMENT:HCPM:MINIMUM:STATUS?` might return `PASS`, `FAIL`, or `NA`.

FETCH:P25:SUMMary:POWEr:MEASurement:HCPM:OFFSlot? (Query Only)

Returns the HCPM Offslot power results.

Conditions

Measurement view: P25 Summary

Group

Fetch commands

Syntax

`FETCH:P25:SUMMary:POWEr:MEASurement:HCPM:OFFSlot?`

Returns

HCPM Offslot power results.

Examples

`FETCH:P25:SUMMARY:POWER:MEASUREMENT:HCPM:OFFSLOT?` would return the HCPM Offslot power result.

FETCh:P25:SUMMary:POWEr:MEASurement:HCPM:OFFSlot:STATus? (Query Only)

Returns Status of HCPM Offslot measurement based on the limit.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Fetch commands

Syntax

`FETCh:P25:SUMMary:POWEr:MEASurement:HCPM:OFFSlot:STATus?`

Arguments

None

Examples

`FETCH:P25:SUMMARY:POWER:MEASUREMENT:HCPM:OFFSLOT:STATUS?` might return `PASS`, `FAIL`, or `NA`.

FETCh:P25:SUMMary:POWEr:MEASurement:HCPM:PEAK:ACPR:HI (Query Only)

Returns P_ACP_HI value which is displayed on P25 Summary display.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Fetch commands

Syntax

`FETCh:P25:SUMMary:POWEr:MEASurement:HCPM:PEAK:ACPR:HI?`

Arguments

None

Examples

`FETCH:P25:SUMMARY:POWER:MEASUREMENT:HCPM:PEAK:ACPR:HI?` might return `-37.963`, indicating the P_ACP_HI value is -37.96 dBm.

FETCh:P25:SUMMary:POWEr:MEASurement:HCPM:PEAK:ACPR:LOW (Query

Returns P_ACP_LOW value which is displayed on P25 Summary display.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Fetch commands

Syntax

`FETCh:P25:SUMMary:POWEr:MEASurement:HCPM:PEAK:ACPR:LOW`

Arguments

None

Examples

`FETCH:P25:SUMMARY:POWER:MEASUREMENT:HCPM:PEAK:ACPR:LOW?` might return `-40.125`, indicating P_ACP_LOW value is -40.13 dBm.

FETCh:P25:SUMMary:POWEr:MEASurement:HCPM:PEAK:ACPR:MIN (Query

Returns Min Pk ACPR value which is displayed on P25 Summary Display.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Fetch commands

Syntax

`FETCh:P25:SUMMary:POWEr:MEASurement:HCPM:PEAK:ACPR:MIN`

Arguments

None

Examples

`FETCH:P25:SUMMARY:POWER:MEASUREMENT:HCPM:PEAK:ACPR:MIN?` might return `47.96`, indicating Min Pk ACPR value is 47.96 dBm

FETCh:P25:SUMMary:POWEr:MEASurement:HCPM:PEAK:ACPR:MIN:STATus

Returns status of Min Pk ACPR value based on the limit.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Fetch commands

Syntax

`FETCh:P25:SUMMary:POWEr:MEASurement:HCPM:PEAK:ACPR:MIN:STATus`

Arguments

None

Examples

`FETCH:P25:SUMMARY:POWER:MEASUREMENT:HCPM:PEAK:ACPR:MIN:STATUS?` might return `PASS`, `FAIL`, or `NA`.

FETCh:P25:SUMMary:POWEr:MEASurement:RF (Query Only)

Returns RF measurement results.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Fetch commands

Syntax

`FETCh:P25:SUMMary:POWEr:MEASurement:RF`

Arguments

None

Examples

`FETCH:P25:SUMMARY:POWER:MEASUREMENT:RF`

FETCh:P25:SUMMary:POWEr:MEASurement:RF:STATus (Query Only)

Returns Status of RF measurement status.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Fetch commands

Syntax

```
FETCh:P25:SUMMary:POWEr:MEASurement:RF:STATus
```

Arguments

None

Returns

PASS, FAIL, or NA.

Examples

`FETCH:P25:SUMMARY:POWER:MEASUREMENT:RF:STATUS` might return `PASS`, indicating that PASS is the status result of the measurement.

FETCh:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ONE:.ACQ:COUNT (Query Only)

Returns HCPM t_error_1 acquisition count.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Fetch commands

Syntax

```
FETCh:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ONE:.ACQ:COUNT?
```

Arguments

None

Examples

```
FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:HCPM:TIME:ALIGNMENT:TERROR:ONE:.ACQ:COUNT?
```

FETCh:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ONE:ACQ:COUNT (Query Only)

Returns the HCPM t_error_1 acquisition count in P25 Summary display.

This is applicable when phase 2 is selected as the standard in the Modulation Params tab.

Conditions

Measurement view: P25 Summary

Group

Fetch commands

Syntax

FETCh:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment: TERRor:ONE:ACQ:COUNT?

Arguments

None

Returns

<NR1> Range is 0 to 5

Examples

FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:HCPM:TIME:ALIGNMENT:TERROR:ONE:ACQ:COUNT might return 1, indicating HCPM t_error_1 acquisition count is 1.

FETCh:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ZERO:STATus (Query Only)

Queries the status of the HCPM Time alignment's t_error_0 measurement results in the P25 Summary display .

This is applicable when phase 2 is selected as the standard in the Modulation Params tab.

Conditions

Measurement view: P25 Summary

Group

Fetch commands

Syntax

FETCh:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ZERO:STATus?

Arguments

None

Returns

<String>

PASS indicates that the test passed.

FAIL indicates that the test failed.

NA indicates that the test is not Applicable.

Examples

FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:HCPM:TIME:ALIGNMENT:TERROR:ZERO:STATUS? might return Pass, indicating that the HCPM Time alignment's t_error_0 measurement test result is Pass.

FETCh:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ONE:STATus (Query Only)

Returns the Status of HCPM Time alignment's t_error_1 measurement results in P25 Summary display.

This is applicable when phase 2 is selected as the standard in the Modulation Params tab.

Conditions

Measurement view: P25 Summary

Group

Fetch commands

Syntax

`FETCh:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ONE:STATus?`

Arguments

None

Returns

<String>

PASS indicates that the test passed.

FAIL indicates that the test failed.

NA indicates that the test is not Applicable.

Examples

`FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:HCPM:TIME:ALIGNMENT:TERROR: ONE:STATUS?` might return `Pass`, indicating that the HCPM Time alignment's t_error_1 measurement test result is Pass.

FETCh:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ZERO:ACQ:COUNT (Query Only)

Returns the HCPM t_error_0 acquisition count in P25 Summary display.

This is applicable when phase 2 is selected as the standard in the Modulation Params tab.

Conditions

Measurement view: P25 Summary

Group

Fetch commands

Syntax

`FETCh:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ZERO:ACQ:COUNT?`
`Arguments`

None

Returns

<NR1> Range is 0 to 5

Examples

`FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:HCPM:TIME:ALIGNMENT:TERROR:ZERO:ACQ:COUNT` might return `3`, indicating HCPM t_error_0 acquisition count is 3.

FETCh:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:.ZERO:S(Query Only)

Returns status of HCPM Time alignment's t_error_0 measurement results.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Fetch commands

Syntax

`FETCh:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:.ZERO:STATus?`

Arguments

None

Examples

`FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:HCPM:TIME:ALIGNMENT:TERROR:.ZERO:STATUS?` might return `PASS`, `FAIL`, or `NA`.

FETCH:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ONE:.ACQ:COUNT?(Query Only)

Returns the HCPM Time alignment's t_error_1 acquisition count over which the t_error_1 average is computed.

Conditions

Measurement view: P25 Summary

Group

Fetch commands

Syntax

`FETCH:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ONE:.ACQ:COUNT?`

Returns

HCPM Time alignment's t_error_1 acquisition count.

Examples

`FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:HCPM:TIME:ALIGNMENT:TERROR:ONE:.ACQ:COUNT?`

FETCH:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ONE:?(Query Only)

Returns the HCPM Time alignment's t_error_1 measurement results.

Conditions

Measurement view: P25 Summary

Group

Fetch commands

Syntax

FETCH:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ONE?

Returns

HCPM Time alignment's t_error_1 measurement results.

Examples

FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:HCPM:TIME:ALIGNMENT:TERROR:ONE?

FETCH:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ZERO?

Returns the HCPM Time alignment's t_error_0 measurement results.

Conditions

Measurement view: P25 Summary

Group

Fetch commands

Syntax

FETCH:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ZERO?

Returns

HCPM Time alignment's t_error_0 measurement results.

Examples

FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:HCPM:TIME:ALIGNMENT:TERROR:ZERO?

FETCH:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TOBSync? (Query Only)

Returns the HCPM Time alignment's t_obsync measurement results.

Conditions

Measurement view: P25 Summary

Group

Fetch commands

Syntax

FETCH:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TOBSync?

Returns

HCPM Time alignment's t_obsync measurement results.

Examples

FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:HCPM:TIME:ALIGNMENT:TOBSYNC?

FETCh:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TOBSync:ACQ:COUNT (Query Only)

Returns the HCPM Time alignment's tob_sync acquisition count in P25 Summary display.

This is applicable when phase 2 is selected as the standard in the Modulation Params tab.

Conditions

Measurement view: P25 Summary

Group

Fetch commands

Syntax

```
FETCh:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TOBSync:ACQ:COUNT?
```

Arguments

None

Returns

<NR1> Range is 0 to 5

Examples

`FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:HCPM:TIME:ALIGNMENT:TOBSYNC:ACQ:COUNT?` might return `2`, indicating HCPM Time alignment's tob_sync acquisition count is 2.

FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:.ENCOder? (Query Only)

Returns the Average Transmitter Encoder Attack Time measurement results.

Conditions

Measurement view: P25 Summary

Group

Fetch commands

Syntax

```
FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:.ENCOder?
```

Returns

Average Transmitter Encoder Attack Time measurement results.

Examples

```
FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:.ENCODER?
```

FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:.ENCOder:BI:STAT(Query Only)

Returns status of Average Transmitter Power Encoder Attack Time, busy/idle, measurement status.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Fetch commands

Syntax

`FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:.ENCOder:BI:STATus?`

Arguments

None

Returns

Average Transmitter Encoder Attack Time , busy/idle, measurement results.

Examples

`FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:.ENCODER:BI:STATUS?` might return `PASS`, `FAIL`, or `NA`.

FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ENCOder: STATus (Query Only)

Returns the Status of Average Transmitter Encoder Attack Time measurement results in P25 Summary display.

This is applicable when phase 1 is selected as the standard in the Modulation Params tab.

Conditions

Measurement view: P25 Summary

Group

Fetch commands

Syntax

`FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ENCOder:STATus?`

Arguments

None

Returns

<String>

PASS indicates that the test passed.

FAIL indicates that the test failed.

NA indicates that the test is not Applicable.

Examples

`FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:ENCODER:STATUS?` might return `Pass`, indicating that Average Transmitter Encoder Attack Time measurement test result is Pass.

FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr:BI (Query Only)

Returns Average Transmitter Power Attack Time, busy/idle measurement results.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Fetch commands

Syntax

```
FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr:BI
```

Arguments

None

Examples

```
FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:POWER:BI
```

FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ACQ:COUNT(Query Only)

Returns Average Transmitter Power Attack acquisition count.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Fetch commands

Syntax

```
FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ACQ:COUNT?
```

Arguments

None

Examples

`FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:ACQ:COUNT?` might return `PASS`, `FAIL`, or `NA`.

FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ACQ:COUNT:BI(Query Only)

Returns the Average Transmitter Power Attack, busy/idle count in the P25 Summary display.

This is applicable when phase 1 is selected as the standard in the Modulation Params tab.

Conditions

Measurement view: P25 Summary

Group

Fetch commands

Syntax

`FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ACQ:COUNT:BI?`

Arguments

None

Returns

<NR1>Range is 0 to 10

Examples

`FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:ACQ:COUNT:BI?` might return `8`, indicating Average Transmitter Power Attack,busy/idle count is 8.

FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ENCOder:BI:STATus (Query Only)

Returns the Status of Average Transmitter Encoder Attack Time (Busy/Idle) measurement results in P25 Summary display.

This is applicable when phase 1 is selected as the standard in the Modulation Params tab.

Conditions

Measurement view: P25 Summary

Group

Fetch commands

Syntax

`FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ENCOder: BI:STATus?`

Arguments

None

Returns

<String>

PASS indicates that the test passed.

FAIL indicates that the test failed.

NA indicates that the test is not Applicable.

Examples

`FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:ENCODER:BI:STATUS?` might return `Pass`, indicating that the Average Transmitter Encoder Attack Time (Busy/Idle) measurement test result was Pass.

FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ENCOder: STATus (Query Only)

Returns the Status of Average Transmitter Encoder Attack Time measurement results in P25 Summary display.

This is applicable when phase 1 is selected as the standard in the Modulation Params tab.

Conditions

Measurement view: P25 Summary

Group

Fetch commands

Syntax

`FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ENCOder:STATus?`

Arguments

None

Returns

<String>

PASS indicates that the test passed.

FAIL indicates that the test failed.

NA indicates that the test is not Applicable.

Examples

`FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:ENCODER:STATUS?` might return `Pass`, indicating that Average Transmitter Encoder Attack Time measurement test result is Pass.

FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ENCOder?(Query Only)

Returns the Average Transmitter Encoder Attack Time measurement results.

Conditions

Measurement view: P25 Summary

Group

Fetch commands

Syntax

`FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ENCOder?`

Returns

Average Transmitter Encoder Attack Time measurement results.

Examples

`FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:ENCODER?`

FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ENCOder:BI?(Query Only)

Returns the Average Transmitter Encoder Attack Time, busy/idle, measurement results.

Conditions

Measurement view: P25 Summary

Group

Fetch commands

Syntax

```
FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ENCOder:BI?
```

Returns

Average Transmitter Encoder Attack Time , busy/idle, measurement results.

Examples

```
FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:ENCODER:BI?
```

FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr (Query Only)

Returns Average Transmitter Power Attack Time measurement results.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Fetch commands

Syntax

```
FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr
```

Arguments

None

Examples

```
FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:POWER
```

FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr:BI (Query Only)

Returns Average Transmitter Power Attack Time, busy/idle measurement results.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Fetch commands

Syntax

```
FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr:BI
```

Arguments

None

Examples

```
FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:POWER:BI
```

FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr:BI:STATus(Query Only)

Returns status of Average Transmitter Power Attack Time, busy/idle, measurement results based on the limit.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Fetch commands

Syntax

```
FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr:BI:STATus
```

Arguments

None

Examples

`FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:POWER:BI:STATUS` might return PASS, FAIL or NA.

FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr:BI?(Query Only)

Returns the Average Transmitter Power Attack Time, busy/idle, measurement results.

Conditions

Measurement view: P25 Summary

Group

Fetch commands

Syntax

```
FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr:BI?
```

Returns

Average Transmitter Power Attack Time, busy/idle, measurement results.

Examples

```
FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:POWER:BI?
```

FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME: POWEr:STATus (Query Only)

Returns status of Average Transmitter Power Attack Time measurement results.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Fetch commands

Syntax

```
FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr:STATus
```

Arguments

None

Examples

`FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:POWER:STATUS` might return `PASS`, `FAIL`, or `NA`.

FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr?(Query Only)

Returns the Average Transmitter Power Attack Time measurement results.

Conditions

Measurement view: P25 Summary

Group

Fetch commands

Syntax

```
FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr?
```

Returns

Average Transmitter Power Attack Time measurement results.

Examples

```
FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:POWER?
```

FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy (Query Only)

Returns Throughput delay measurement results.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Fetch commands

Syntax

```
FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy
```

Arguments

None

Examples

```
FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:THRUPUT:DELAY
```

FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy:ACQ:COUNT (Query Only)

Returns Throughput delay acquisition counts.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Fetch commands

Syntax

```
FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy:ACQ:COUNT
```

Arguments

None

Examples

```
FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:THRUPUT:DELAY:ACQ:COUNT
```

FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy:ACQ:COUNT?(Query Only)

Returns the Throughput delay acquisition counts over which the Throughput delay average is computed.

Conditions

Measurement view: P25 Summary

Group

Fetch commands

Syntax

```
FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy:ACQ:COUNT?
```

Returns

Throughput delay acquisition counts.

Examples

```
FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:THRUPUT:DELAY:ACQ:COUNT?
```

FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy:STATus (Query Only)

Returns status of Throughput delay measurement results based on the limit.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Fetch commands

Syntax

```
FETCh:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy:STATus
```

Arguments

None

Examples

`FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:THRUPUT:DELAY:STATUS` might return `PASS`, `FAIL`, or `NA`.

FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy? (Query Only)

Returns the Throughput delay measurement results.

Conditions

Measurement view: P25 Summary

Group

Fetch commands

Syntax

```
FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy?
```

Returns

Throughput delay measurement results.

Examples

`FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:THRUPUT:DELAY?`

FETCh:PERRor[:MEASview<y>]:FERRor? (Query only)

Returns the frequency error in the Phase error versus Time measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Phase error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Phase error versus Time).

Group

Fetch commands

Syntax

`FETCh:PERRor[:MEASview<y>]:FERRor?`

Arguments

None

Returns

`<freq_error>::<NRf>` is the frequency error in Hz.

Examples

`FETCh:PERRor:MEASview2:FERRor?` might return -10.7E+3, indicating the frequency error is -10.7 kHz for measurement view2.

FETCh:PERRor[:MEASview<y>]:PEAK? (Query only)

Returns the peak value in the Phase error versus Time measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Phase error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Phase error versus Time).

Group

Fetch commands

Syntax

`FETCh:PERRor[:MEASview<y>]:PEAK?`

Related Commands

FETCh:PERRor[:MEASview<y>]:PINDex? (on page 1290)

Arguments

None

Returns

<peak>::<NRf> is the peak phase error in percent (%).

Examples

FETCh:PERRor:MEASview2:PEAK? might return 1.57, indicating the peak phase error value is 1.57% for measurement view 2.

FETCh:PERRor[:MEASview<y>]:PINDex? (Query only)

Returns the time at the phase error peak.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Phase error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Phase error versus Time).

Group

Fetch commands

Syntax

```
FETCh:PERRor[:MEASview<y>]:PINDex?
```

Related Commands

FETCh:PERRor[:MEASview<y>]:PEAK? (on page 1289)

Arguments

None

Returns

<peak_time>::<NRf> is the time at the phase error peak in symbol number.

The unit can be changed by [SENSe]:DDEMod:TIMe:UNITs command.

Examples

FETCh:PERRor:MEASview2:PINDex? might return 68.000, indicating that the Phase error peak is at symbol #68 for measurement view 2.

FETCh:PERRor[:MEASview<y>]:RMS? (Query only)

Returns the RMS (Root-Mean-Square) value in the Phase error versus Time measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Phase error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Phase error versus Time).

Group

Fetch commands

Syntax

`FETCh:PERRor[:MEASview<y>]:RMS?`

Arguments

None

Returns

`<rms>::<NRf>` is the RMS phase error in percent (%).

Examples

`FETCh:PERRor:MEASview2:RMS?` might return 0.382, indicating the RMS Phase error value is 0.382% for measurement view 2.

FETCh:PERRor[:MEASview<y>]:TRACe? (Query only)

Returns the Phase error versus Time trace data.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Phase error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Phase error versus Time).

Group

Fetch commands

Syntax

`FETCh:PERRor[:MEASview<y>]:TRACe?`

Arguments

None

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the phase error in degrees at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

FETCh:PERRor:MEASview2:TRACe? might return #42036xxxx... (2036-byte data) for the Phase error versus Time Measview 2.

FETCh:PHVTime[:MEASview<y>]? (Query Only)

Returns the Phase versus Time trace data.

Conditions

Measurement views: Phase versus Time

Group

Fetch commands

Syntax

FETCh:PHVTime[:MEASview<y>]?

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

#<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the phase in degrees at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

FETCH:PHVTIME:MEASview1? might return #3160xxxx... (160-byte data) for the Phase versus Time trace.

FETCh:PHVTime[:MEASview<y>]:MAXimum? (Query Only)

Returns the maximum value in the Phase versus Time measurement.

Conditions

Measurement views: Phase versus Time

Group

Fetch commands

Syntax

FETCh:PHVTime[:MEASview<y>]:MAXimum?

Related Commands

FETCh:PHVTime[:MEASview<y>]:MAXLocation? (on page 1293)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

<max>::<NRf> is the maximum phase in degrees.

Examples

FETCH:PHVTIME:MEASview1:MAXIMUM? might return 153.8, indicating the maximum phase is 153.8°.

FETCh:PHVTime[:MEASview<y>]:MAXLocation? (Query Only)

Returns the time at which the phase is maximum.

Conditions

Measurement views: Phase versus Time

Group

Fetch commands

Syntax

FETCh:PHVTime[:MEASview<y>]:MAXLocation?

Related Commands

FETCh:PHVTime[:MEASview<y>]:MAXimum? (on page 1292)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

<max_time>::<NRf> is the time in seconds at which the phase is maximum.

Examples

FETCH:PHVTIME:MEASview1:MAXLOCATION? might return 175.3E-9, indicating the I or Q level is maximum at 175.3 ns.

FETCh:PHVTime[:MEASview<y>]:MINimum? (Query Only)

Returns the minimum value in the Phase versus Time measurement.

Conditions

Measurement views: Phase versus Time

Group

Fetch commands

Syntax

FETCh:PHVTime[:MEASview<y>]:MINimum?

Related Commands

FETCh:PHVTime[:MEASview<y>]:MINLocation? (on page 1294)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

<min>::<NRf> is the minimum phase in degrees.

Examples

FETCH:PHVTIME:MEASview1:MINIMUM? might return -176.3, indicating the minimum phase is -176.3°.

FETCh:PHVTime[:MEASview<y>]:MINLocation? (Query Only)

Returns the time at which the phase is minimum.

Conditions

Measurement views: Phase versus Time

Group

Fetch commands

Syntax

FETCh:PHVTime[:MEASview<y>]:MINLocation?

Related Commands

FETCh:PHVTime[:MEASview<y>]:MINimum? (on page 1293)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

<min_time>::<NRf> is the time in seconds at which the phase is minimum.

Examples

FETCH:PHVTIME:MEASview1:MINLOCATION? might return 450.7E-9, indicating the phase is minimum at 450.7 ns.

FETCh:PHVTime[:MEASview<y>]:OGRAm? (Query only)

Returns trace data of a line in the ogram. The line is selected using the TRACe:PHVTime[:MEASview<y>]:OGRAm:SELect:LINE command.

Conditions

Applicable measurement view must be active.

Group

Fetch commands

Syntax

FETCh:PHVTime[:MEASview<y>]:OGRAm?

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional. By default y is 1.

Returns

#<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the phase at the n^{th} data point in degrees (deg), represented as a 4-byte little-endian IEEE 488.2 floating-point value.

Examples

FETCh:PHVTime:MEASview1:OGRAm? might return #43204xxxx... (3204-byte data) for the line in the ogram.

FETCh:PHVTime[:MEASview<y>]:RESult? (Query Only)

Returns the Phase versus Time measurement results.

Conditions

Measurement views: Phase versus Time

Group

Fetch commands

Syntax

FETCh:PHVTime[:MEASview<y>]:RESult?

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

<max>,<max_time>,<min>,<min_time>

Where

<max>::<NRf> is the maximum phase in degrees.

<max_time>::<NRf> is the time in seconds at which the phase is maximum.

<min>::<NRf> is the minimum phase in degrees.

<min_time>::<NRf> is the time in seconds at which the phase is minimum.

Examples

FETCH:PHVTIME::MEASview1:RESULT? might return 153.8,175.3E-9,-176.3,450.7E-9, indicating the maximum phase is 153.8 ° at 175.3 ns and the minimum phase is -176.3 ° at 450.7 ns.

FETCh:PHVTime[:MEASview<y>]:TRACe:BITMap? (Query Only)

Returns trace waveform data of the bitmap trace in the Phase versus Time measurement.

Conditions

Measurement views: Phase versus Time

Group

Fetch commands

Syntax

`FETCh:PHVTime[:MEASview<y>]:TRACe:BITMap?`

Arguments

None

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Returns

A binary block of 32-bit floating point values (IEEE standard single precision floating point format). Each value ranges from 0 to 1.0, where 1.0 represents 100%.

Bitmaps have 201 rows. Data[0] is the first column in the first row, and the following data continues along that row before moving to the next.

Bitmap values are returned in the order of top left pixel first, then the rest of the top row, after that row by row.
In other words, the pixels in the bitmap are returned from left to right and top to bottom; one row at a time.

Examples

`FETCH:PHVTIME:TRACE:BITMAP?` might return #6644004xxxx...

FETCh:PM:PNEGative? (Query Only)

Returns the negative peak phase deviation (–Pk) in the PM measurement.

Conditions

Measurement views: PM

Group

Fetch commands

Syntax

`FETCh:PM:PNEGative?`

Arguments

None

Returns

`<-Pk>::<NRf>` is the negative peak phase deviation in degrees.

Examples

FETCH:PM:PNEGATIVE? might return -23.42, indicating the positive peak phase deviation is -23.42 °.

FETCh:PM:PPOSitive? (Query Only)

Returns the positive peak phase deviation (+Pk) in the PM measurement.

Conditions

Measurement views: PM

Group

Fetch commands

Syntax

FETCh:PM:PPOSitive?

Arguments

None

Returns

<+Pk>::<NRf> is the positive peak phase deviation in degrees.

Examples

FETCH:PM:PPOSITIVE? might return 26.87, indicating the positive peak phase deviation is 26.87 °.

FETCh:PM:PTPeak? (Query Only)

Returns the peak-peak phase deviation (Pk-Pk) in the PM measurement.

Conditions

Measurement views: PM

Group

Fetch commands

Syntax

FETCh:PM:PTPeak?

Arguments

None

Returns

<Pk-Pk>::<NRf> is the peak-peak phase deviation in degrees.

Examples

FETCH:PM:PTPEAK? might return 46.34, indicating the peak-peak phase deviation is 46.34 °.

FETCh:PM:RESult? (Query Only)

Returns the PM measurement results.

Conditions

Measurement views: PM

Group

Fetch commands

Syntax

`FETCh:PM:RESult?`

Arguments

None

Returns

`<+Pk>,<-Pk>,<RMS>,<Pk-Pk>`

Where

`<+Pk>::<NRf>` is the positive peak phase deviation in degrees.

`<-Pk>::<NRf>` is the negative peak phase deviation in degrees.

`<RMS>::<NRf>` is the RMS phase deviation in degrees.

`<Pk-Pk>::<NRf>` is the peak-peak phase deviation in degrees.

Examples

`FETCH:PM:RESULT?` might return `22.89,-23.45,15.12,46.34`.

FETCh:PM:RMS? (Query Only)

Returns the RMS phase deviation in the PM measurement.

Conditions

Measurement views: PM

Group

Fetch commands

Syntax

`FETCh:PM:RMS?`

Arguments

None

Returns

`<RMS>::<NRf>` is the RMS phase deviation in degrees.

Examples

`FETCH:PM:RMS?` might return 15.12, indicating the RMS frequency deviation is 15.12 °.

FETCh:PNOise:ALL? (Query Only)

Returns all results of the phase noise measurement.

Conditions

Measurement views: Phase noise

Group

Fetch commands

Syntax

`FETCh:PNOise:ALL?`

Arguments

None

Returns

`<Cpower>,<Ferror>,<Pnoise>,<Tjitter>,<Rjitter>,<RFM>`

Where

`<Cpower>::<NRf>` is the carrier power in dBm.

`<Ferror>::<NRf>` is the frequency error in Hz.

`<Pnoise>::<NRf>` is the RMS phase noise in degrees.

`<Tjitter>::<NRf>` is the total jitter in seconds.

`<Rjitter>::<NRf>` is the random jitter in seconds.

`<RFM>::<NRf>` is the residual FM in Hz.

Examples

`FETCH:PNOISE:ALL?` might return -9.455,1.235E+6,51.43,2.312E-9,4.178E-9,14.58, indicating Carrier power: -9.455 dBm, Frequency error: 1.235 MHz, RMS phase noise: 51.43 °, Total jitter: 2.312 ns, Random jitter: 4.178 ns, and Residual FM: 14.58 Hz.

FETCh:PNOise:CARRier:FERRor? (Query Only)

Returns the carrier frequency error in the phase noise measurement.

Conditions

Measurement views: Phase noise

Group

Fetch commands

Syntax

`FETCh:PNOise:CARRier:FERRor?`

Arguments

None

Returns

`<NRf>` Carrier frequency error in Hz.

Examples

FETCH:PNOISE:CARRIER:FERROR? might return 1.235E+6, indicating that the carrier frequency error is 1.235 MHz.

FETCh:PNOise:CARRier:POWer? (Query Only)

Returns the carrier power in the phase noise measurement.

Conditions

Measurement views: Phase noise

Group

Fetch commands

Syntax

FETCh:PNOise:CARRier:POWer?

Arguments

None

Returns

<NRf> Carrier power in dBm.

The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples

FETCH:PNOISE:CARRIER:POWER? might return -9.455, indicating that the carrier power is -9.455 dBm.

FETCh:PNOise:JITTer? (Query Only)

Returns the jitter in the phase noise measurement.

Conditions

Measurement views: Phase noise

Group

Fetch commands

Syntax

FETCh:PNOise:JITTer?

Arguments

None

Returns

<NRf> Jitter in seconds.

Examples

FETCH:PNOISE:JITTER? might return 2.312E-9, indicating that the jitter is 2.312 ns.

FETCh:PNOise:RESidual:FM? (Query Only)

Returns the residual FM in the phase noise measurement.

Conditions

Measurement views: Phase noise

Group

Fetch commands

Syntax

`FETCh:PNOise:RESidual:FM?`

Arguments

None

Returns

`<NRf>` Residual FM in Hz.

Examples

`FETCH:PNOISE:RESIDUAL:FM?` might return 14.58, indicating that the residual FM is 14.58 Hz.

FETCh:PNOise:RMS:PNOise? (Query Only)

Returns the RMS phase noise in the phase noise measurement.

Conditions

Measurement views: Phase noise

Group

Fetch commands

Syntax

`FETCh:PNOise:RMS:PNOise?`

Arguments

None

Returns

`<NRf> RMS` phase noise in degrees.

Examples

`FETCH:PNOISE:RMS:PNOISE?` might return 51.43, indicating that the RMS phase noise is 51.43 °.

FETCh:PNOise:SPECtrum<x>:X? (Query Only)

Returns the frequencies of the specified trace.

The parameter <x> = 1 and 2, representing Trace 1 and Trace 2, respectively.

Conditions

Measurement views: Phase noise

Group

Fetch commands

Syntax

`FETCh:PNOise:SPECtrum<x>:X?`

Arguments

None

Returns

`#<num_digit><num_byte><x(1)><x(2)>...<x(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<x(n)>` is the frequency (Hz) at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`FETCH:PNOISE:SPECTRUM1:X?` might return #516020xxxx... (16020-byte data) for the frequencies of Trace 1.

FETCh:PNOise:SPECtrum<x>:XY? (Query Only)

Returns the frequency and phase noise pairs of the specified trace.

The parameter <x> = 1 and 2, representing Trace 1 and Trace 2, respectively.

Conditions

Measurement views: Phase noise

Group

Fetch commands

Syntax

`FETCh:PNOise:SPECtrum<x>:XY?`

Arguments

None

Returns

`#<num_digit><num_byte><x(1)><y(1)><x(2)><y(2)>...<x(n)><y(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<x(n)><y(n)>` is the frequency (Hz) and phase noise (dBc/Hz) pair at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`FETCH:PNOISE:SPECTRUM1:XY?` might return #516020xxxx... (16020-byte data) for the frequency and phase noise pairs of the Trace 1.

FETCh:PNOise:SPECtrum<x>[:Y]? (Query Only)

Returns the phase noise values of the specified trace.

The parameter <x> = 1 and 2, representing Trace 1 and Trace 2, respectively.

Conditions

Measurement views: Phase noise

Group

Fetch commands

Syntax

`FETCh:PNOise:SPECtrum<x>[:Y]?`

Arguments

None

Returns

`#<num_digit><num_byte><y(1)><y(2)>...<y(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<y(n)>` is the phase noise (dBc/Hz) at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`FETCH:PNOISE:SPECTRUM1:Y` might return `#516020xxxx...` (16020-byte data) for the phase noise values of Trace 1.

FETCh:PULSe:CUMulative:HISTogram[:MEASview<y>]:OUTLier:HIGHer? (Query Only)

Returns the outliers value exceeding the upper limit in pulse cumulative histogram.

Conditions

Measurement views: Cumulative Histogram with the specified Measurement View Number

Group

Fetch commands

Syntax

`FETCh:PULSe:CUMulative:HISTogram[:MEASview<y>]:OUTLier:HIGHer?`

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<NRf>` Number of outliers above higher limit

Examples

FETCH:PULSE:CUMULATIVE:HISTOGRAM:MEASview1:OUTLIER:HIGHER? might return 0.000, which is the value for the outliers that have exceeded the upper limit in the pulse cumulative histogram.

FETCh:PULSe:CUMulative:HISTogram[:MEASview<y>]:OUTLier:LOWer? (Query Only)

Returns the outliers value below the lower limit in pulse cumulative histogram.

Conditions

Measurement views: Cumulative Histogram with the specified Measurement View Number

Group

Fetch commands

Syntax

FETCh:PULSe:CUMulative:HISTogram[:MEASview<y>]:OUTLier:LOWer?

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf> Number of outliers below lower limit

Examples

FETCH:PULSE:CUMULATIVE:HISTOGRAM:MEASview1:OUTLIER:LOWER? might return 0.000, which is the value for the outliers below the lower limit in the pulse cumulative histogram.

FETCh:PULSe:CUMulative:HISTogram[:MEASview<y>]:PULse:COUNt? (Query Only)

Returns the number of pulses in the bin in the pulse cumulative histogram.

Conditions

Measurement views: Pulse Cumulative Histogram

Group

Fetch commands

Syntax

FETCh:PULSe:CUMulative:HISTogram[:MEASview<y>]:PULse:COUNt?

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf> Number of pulses in the bin

Examples

FETCH:PULSE:CUMULATIVE:HISTOGRAM:MEASview1:PULSE:COUNT? might return 0 for the number of pulses in the bin.

FETCh:PULSe:CUMulative:HISTogram[:MEASview<y>]:PULse:TOTal? (Query Only)

Returns the total number of pulses in the pulse cumulative histogram.

Conditions

Measurement views: Cumulative Histogram with the specified Measurement View Number

Group

Fetch commands

Syntax

FETCh:PULSe:CUMulative:HISTogram[:MEASview<y>]:PULse:TOTal?

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf> Total number of pulses

Examples

FETCH:PULSE:CUMULATIVE:HISTOGRAM:MEASview1:PULSE:TOTAL? might return 16 for total number pulses in the pulse cumulative histogram.

FETCh:PULSe:CUMulative:STATistics[:MEASview<y>]

Returns the PulseCount, Max, Max TimeStamp, Min, Min TimeStamp, Peak to Peak, Average and Standard Deviation values for the chosen measurement in the pulse cumulative statistics display.

Conditions

Measurement views: Pulse Cumulative Statistics

Group

Fetch commands

Syntax

FETCh:PULSe:CUMulative:STATistics[:MEASview<y>] <arg1>,<arg2>

Arguments

arg1 can be one of the following measurements:

{AVERage | PPOWer | ATX | WIDTh | RISE | FALL | RINTerval | RRATe | DUTPct | DUTRatio | RIPPle | RIPDb | DROop | DRODb | OVERshoot | OVEDb| PPPHase | PPFRequency | RMSFreqerror | MFReqerror | RMSPherror |MPHerror | FRDeviation | FDELta | PHDeviation | IRAMplitude | IRTime | PPPD| PPFD | FABS}

arg2 can be one of the following statistics results:

{TPCOunt|MAX|MAXTs|MIN|MINTs|PK2PK|AVG|SD|ALL}

Returns

<puls_cnt>|<max>|<max_timstmp>|<min>|<min_timstmp>|pk_pk>|std_dev>|< for one of the following selected measurements:

<avg_on> | <pk_pwr> | <avg_tx> | <width> | <rise> | <fall> | <rep_int> | <rep_rate> | <duty_factor_pcnt> | <duty_fact_ratio> | <ripple> | <ripple_db> | <droop> | <droop_db> | <ovrshoot> | <ovrshoot_db> | <pp_phase> | <pp_freq><rms_phase_err> | <rms_freq_err> | <p2p_phase> | <p2p_freq> | <pk_phase_err> | <pk_freq_err> | <abs_freq> | <freq_dev><phase_dev> | <dlta_freq> | <imp_resp_amp> | <imp_resp_time>

Where

<avg_on>::<NRf> is the average power transmitted during pulse ON.

<pk_pwr>::<NRf> is the Maximum power during pulse on.

<pk_pwr>::<NRf> is the average power transmitted.

<width>::<NRf> is the time from the rising edge to the falling edge at the –3 dB/–6 dB level (50%) of the user selected 100% level.

<rise>::<NRf> is the time required for a signal to rise from 10% to 90% (or 20% to 80%) of the user selected 100% level.

<fall>::<NRf> is the time required for a signal to fall from 90% to 10% (or 80% to 20%) of the user selected 100% level.

<rep_int>::<NRf> is the time from a pulse rising edge to the next pulse rising edge.

<rep_rate>::<NRf> is the inverse of repetition interval.

<duty_factor_pcnt>::<NRf> is the ratio of the width to the pulse period, expressed as a percentage.

<duty_fact_ratio>::<NRf> is the ratio of the pulse width to the pulse period.

<ripple>::<NRf> is the Ripple is the peak-to-peak ripple on the pulse top.

<ripple_db>::<NRf> is the Ripple measurement expressed in dB.

<droop>::<NRf> is the Droop is the power difference between the beginning and the end of the pulse On time.

<droop_db>::<NRf> is the Droop measurement expressed in dB.

<ovrshoot>::<NRf> is the amount by which the signal exceeds the 100% level on the pulse rising edge.

<ovrshoot_db>::<NRf> is the Over shoot measurement expressed in dB.

<p2p_phase>::<NRf> is the phase difference between the selected pulse and the previous pulse in the analysis window.

<p2p_freq>::<NRf> is the difference between the frequency of the current pulse and frequency of the previous pulse.

<rms_phase_err>::<NRf> is the RMS average of the Phase vs Time trace, computed over the Measurement Time.

<rms_freq_err>::<NRf> is the RMS average of the Freq Error vs.Time trace.

<pp_phase>::<NRf> is the frequency difference between the current pulse and the first pulse.

<pp_freq>::<NRf> is the frequency difference between the current pulse and immediate previous pulse.

<pk_phase_err>::<NRf> is the maximum phase error.

<abs_freq>::<NRf> is the absolute pulse frequency measured at a specific point.

<freq_dev>::<NRf> is the difference between the maximum and minimum measured values of the signal frequency during the Measurement Time.

<phase_dev>::<NRf> is the difference between the maximum and minimum Phase values measured during the ON time of a pulse.

<dlta_freq>::<NRf> is the difference from the measurement frequency to each pulse frequency.

<imp_resp_amp>::<NRf> is the difference in dB between the levels of the main lobe and highest side lobe.

<imp_resp_time>::<NRf> is the difference in time between the main lobe and highest side lobe.

Examples

FETCh:PULSe:CUMulative:STATistics:MEASview1 AVERage,MAX might return -1.3594852284, which is the max average value for the Average ON measurement.

FETCh:PULSe:CUMulative:STATistics[:MEASview<y>](Query Only)

Returns the PulseCount, Max, Max TimeStamp, Min, Min TimeStamp, Peak to Peak, Average, and Standard Deviation values for the chosen measurement in the Pulse Cumulative Statistics display.

Conditions

Measurement views: Pulse Cumulative Statistics

Group

Fetch commands

Syntax

FETCh:PULSe:CUMulative:STATistics[:MEASview<y>]? <arg1><arg2>

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

<arg1> = {AVERage | PPOWer | ATX | WIDTh | RISE | FALL | RINTerval | RRATe | DUTPct | DUTRatio |RIPPle | RIPDb | DROop | DRODb | OVERshoot | OVEDb | PPPHase | PPFRequency | RMSFreqerror | MFReqerror | RMSPherror |MPHerror | FRDeviation | FDELta | PHDeviation | IRAMplitude | IRTime | PPPD| PPFD | FABS}

<arg2> = {TPCOunt | MAX | MAXTs | MIN | MINTs | PK2PK | AVG | SD | ALL}

Returns

<puls_cnt> | <max> | <max_timstmp> | <min> | <min_timstmp> | <pk_pk> | <avg> | <std_dev> | <all>

For one of the following selected measurements:

<avg_on> | <pk_pwr> | <avg_tx> | <width> | <rise> | <fall> | <rep_int> | <rep_rate> | <duty_factor_pcnt> | <duty_fact_ratio> | <ripple> | <ripple_db> | <droop> | <droop_db> | <ovrshoot> | <ovrshoot_db> | <pp_phase> | <pp_freq> | <rms_phase_err> | <rms_freq_err> | <p2p_phase> | <p2p_freq> | <pk_phase_err> | <pk_freq_err> | <abs_freq> | <freq_dev> | <phase_dev> | <dlta_freq> | <imp_resp_amp> | <imp_resp_time>

where

<avg_on>::<NRf> is the average power transmitted during pulse on.

<pk_pwr>::<NRf> is the Maximum power during pulse on

<avg_tx>::<NRf> is the average power transmitted

<width>::<NRf> is the time from the rising edge to the falling edge at the –3 dB /–6 dB level (50%) of the user selected 100% level

<rise>::<NRf> is the time required for a signal to rise from 10% to 90% (or 20% to 80%) of the user selected 100% level

<fall>::<NRf> is the time required for a signal to fall from 90% to 10% (or 80% to 20%) of the user selected 100% level

<rep_int>::<NRf> is the time from a pulse rising edge to the next pulse rising edge

<rep_rate>::<NRf> is the inverse of repetition interval

<duty_factor_pcnt>::<NRf> is the ratio of the width to the pulse period, expressed as a percentage

<duty_fact_ratio>::<NRf> is the ratio of the pulse width to the pulse period

<ripple>::<NRf> is the Ripple is the peak-to-peak ripple on the pulse top

<ripple_db>::<NRf> is the Ripple measurement expressed in dB

<droop>::<NRf> is the Droop is the power difference between the beginning and the end of the pulse On time

<droop_db>::<NRf> is the Droop measurement expressed in dB

<ovrshoot>::<NRf> is the amount by which the signal exceeds the 100% level on the pulse rising edge

<ovrshoot_db>::<NRf> is the Overshoot measurement expressed in dB

<pp_phase>::<NRf> is the phase difference between the selected pulse and the previous pulse in the analysis window

<pp_freq>::<NRf> is the difference between the frequency of the current pulse and frequency of the previous pulse

<rms_phase_err>::<NRf> is the RMS average of the Phase vs Time trace, computed over the Measurement Time

<rms_freq_err>::<NRf> is the RMS average of the Freq Error vs.Time trace

<p2p_phase>::<NRf> is the frequency difference between the current pulse and the first pulse

<p2p_freq>::<NRf> is the frequency difference between the current pulse and immediate previous pulse

<pk_phase_err>::<NRf> is the maximum phase error

<abs_freq>::<NRf> is the absolute pulse frequency measured at a specific point

<freq_dev>::<NRf> is the difference between the maximum and minimum measured values of the signal frequency during the Measurement Time

<phase_dev>::<NRf> is the Phase Deviation difference between the maximum and minimum Phase values measured during the On time of a pulse

<dlta_freq>::<NRf> is the difference from the measurement frequency to each pulse frequency

<imp_resp_amp>::<NRf> is the difference in dB between the levels of the main lobe and highest side lobe

<imp_resp_time>::<NRf> is the difference in time between the main lobe and highest side lobe

Examples

FETCh:PULSe:CUMulative:STATistics:MEASview1? AVERage,MAX might return -1.3594852284, indicating the max value for Average ON is -1.3594852284.

FETCh:PULSe:OGRAm[:MEASview<y>]:MAX:FRAMe:COUNt (Query Only)

Queries the frame count value in the Pulse-Ogram display.

Conditions

Measurement views: Pulse-Ogram

Group

Fetch commands

Syntax

```
FETCh:PULSe:OGRAm[:MEASview<y>]:MAX:FRAMe:COUNt?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Returns

<NRf> Total number of fast frames

Examples

`FETCh:PULSe:OGRAm:MEASview1:MAX:FRAMe:COUNt?` might return `15`, indicating the total number of fast frames in the acquisition of pulse-Ogram is 15.

FETCh:PULSe:OGRAm[:MEASview<y>]:RBW (Query Only)

Queries the resolution bandwidth value only when Frequency vs Time display is opened in the Pulse-Ogram display.

Conditions

Measurement views: Pulse-Ogram

Group

Fetch commands

Syntax

```
FETCh:PULSe:OGRAm[:MEASview<y>]:RBW?
```

Arguments

`<y>` = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Returns

<NRf>

Examples

`FETCh:PULSe:OGRAm:MEASview1:RBW?` might return `222.9199886322E+3`, indicating the RBW value in Pulse-Ogram is 222.9199886322E+3.

FETCh:PULSe[:RESult][:MEASview<y>]:ADVanced (Query Only)

Queries the PulseCount, Max, Max TimeStamp, Min, Min TimeStamp, Peak to Peak, Average, and Standard Deviation values for the chosen measurement in the Pulse Table display.

Conditions

Measurement views: Pulse Table

Group

Fetch commands

Syntax

FETCh:PULSe[:RESult][:MEASview<y>]:ADVanced <arg1><arg2>

Arguments

<arg1>={Pulse_number}

<arg2>= {AVERage | PPOWer | ATX | WIDTh | RISE | FALL | RINTerval | RRATe | DUTPct | DUTRatio |RIPPle | RIPDb | DROop | DRODb | OVERshoot | OVEDb | PPPHase | PPFRequency | RMSFreqerror | MFReqerror | RMSPherror |MPHerror | FRDeviation | FDELta | PHDeviation | IRAMplitude|TIME | IRTime | PPPD| PPFD | FABS}

Returns

<puls_cnt> | <max> | <max_timstmp> | <min> | <min_timstmp> | <pk_pk> | <avg> | <std_dev> | <all>

For one of the following selected measurements:

<avg_on> | <pk_pwr> | <avg_tx> | <width> | <rise> | <fall> | <rep_int> | <rep_rate> | <duty_factor_pcnt> | <duty_fact_ratio> | <ripple> | <ripple_db> | <droop> | <droop_db> | <ovrshoot> | <ovrshoot_db> | <pp_phase> | <pp_freq> | <rms_phase_err> | <rms_freq_err> | <p2p_phase> | <p2p_freq> | <pk_phase_err> | <pk_freq_err> | <abs_freq> | <freq_dev> | <phase_dev> | <dlta_freq> | <imp_resp_amp> | <imp_resp_time>|<time> for the given pulse number

where

<avg_on>::<NRf> is the average power transmitted during pulse on.

<pk_pwr>::<NRf> is the Maximum power during pulse on

<avg_tx>:: <NRf> is the average power transmitted

<width>::<NRf> is the time from the rising edge to the falling edge at the −3 dB/ −6 dB level (50%) of the user selected 100% level

<rise>::<NRf> is the time required for a signal to rise from 10% to 90% (or 20% to 80%) of the user selected 100% level

<fall> ::<NRf> is the time required for a signal to fall from 90% to 10% (or 80% to 20%) of the user selected 100% level

<rep_int>::<NRf> is the time from a pulse rising edge to the next pulse rising edge

<rep_rate> ::<NRf> is the inverse of repetition interval

<duty_factor_pcnt>::<NRf> is the ratio of the width to the pulse period, expressed as a percentage

<duty_fact_ratio>::<NRf> is the ratio of the pulse width to the pulse period

<ripple>::<NRf> is the Ripple is the peak-to-peak ripple on the pulse top

<ripple_db>::<NRf> is the Ripple measurement expressed in dB

<droop>::<NRf> is the Droop is the power difference between the beginning and the end of the pulse On time

<droop_db>::<NRf> is the Droop measurement expressed in dB

<ovrshoot>::<NRf> is the amount by which the signal exceeds the 100% level on the pulse rising edge

<ovrshoot_db>::<NRf> is the Overshoot measurement expressed in dB

<pp_phase>::<NRf> is the phase difference between the selected pulse and the previous pulse in the analysis window

<pp_freq> ::<NRf> is the difference between the frequency of the current pulse and frequency of the previous pulse

<rms_phase_err>::<NRf> is the RMS average of the Phase vs Time trace, computed over the Measurement Time

<rms_freq_err>::<NRf> is the RMS average of the Freq Error vs.Time trace

<p2p_phase>::<NRf> is the frequency difference between the current pulse and the first pulse

<p2p_freq>::<NRf> is the frequency difference between the current pulse and immediate previous pulse

<pk_phase_err>::<NRf> is the maximum phase error

<abs_freq>::<NRf> is the absolute pulse frequency measured at a specific point

<freq_dev>::<NRf> is the difference between the maximum and minimum measured values of the signal frequency during the Measurement Time

<phase_dev>::<NRf> is the Phase Deviation difference between the maximum and minimum Phase values measured during the On time of a pulse

<dlta_freq>::<NRf> is the difference from the measurement frequency to each pulse frequency

<imp_resp_amp>::<NRf> is the difference in dB between the levels of the main lobe and highest side lobe

<imp_resp_time>::<NRf> is the difference in time between the main lobe and highest side lobe

<time>::<NRf> is the dtime in seconds relative to the time reference point in the first acquisition record in the data set

Examples

FETCH:PULSE[:RESULT]:MEASview1:ADVANCED 1,AVERage might return -1.4063608208, indicating the average ON value for the pulse number 1 is -1.4063608208.

FETCh:PULSe[:RESult][:MEASview<y>]:ADVanced:RANGe (Query only)

Queries the Pulse Count, Max, Max TimeStamp, Min, Min TimeStamp, Peak to Peak, Average, and Standard Deviation values for the selected measurement in the Pulse Table display.

Conditions

Measurement views: Pulse Table

Group

Fetch commands

Syntax

```
FETCh:PULSe[:RESult][:MEASview<y>]:ADVanced:RANGe <arg1><arg2><arg3>
```

Arguments

<arg1>={Start_Pulse_number}

<arg2>={End_Pulse_number}

<arg3>= {AVERage | PPOWer | ATX | WIDTh | RISE | FALL | RINTerval | RRATe | DUTPct | DUTRatio |RIPPle | RIPDb | DROop | DRODb | OVERshoot | OVEDb | PPPHase | PPFRequency | RMSFreqerror | MFReqerror | RMSPherror |MPHerror | FRDeviation | FDELta | PHDeviation | IRAMplitude|TIME | IRTime | PPPD| PPFD | FABS}

Returns

<puls_cnt> | <max> | <max_timstmp> | <min> | <min_timstmp> | <pk_pk> | <avg> | <std_dev> | <all>

For one of the following selected measurements:

<avg_on> | <pk_pwr> | <avg_tx> | <width> | <rise> | <fall> | <rep_int> | <rep_rate> | <duty_factor_pcnt> | <duty_fact_ratio> | <ripple> | <ripple_db> | <droop> | <droop_db> | <ovrshoot> | <ovrshoot_db> | <pp_phase> | <pp_freq> | <rms_phase_err> | <rms_freq_err> | <p2p_phase> | <p2p_freq> | <pk_phase_err> | <pk_freq_err> | <abs_freq> | <freq_dev> | <phase_dev> | <dlta_freq> | <imp_resp_amp> | <imp_resp_time>|<time> for the given pulse number.

where

<avg_on>::<NRf> is the average power transmitted during pulse on.

<pk_pwr>::<NRf> is the Maximum power during pulse on.

<avg_tx>:: <NRf> is the average power transmitted .

<width>::<NRf> is the time from the rising edge to the falling edge at the –3 dB or –6 dB level (50%) of the user selected 100% level.

<rise>::<NRf> is the time required for a signal to rise from 10% to 90% or (20% to 80%) of the user selected 100% level.

<fall> ::<NRf> is the time required for a signal to fall from 90% to 10% or (80% to 20%) of the user selected 100% level.

<rep_int>::<NRf> is the time from a pulse rising edge to the next pulse rising edge.

<rep_rate> ::<NRf> is the inverse of repetition interval.

<duty_factor_pcnt>::<NRf> is the ratio of the width to the pulse period, expressed as a percentage.

<duty_fact_ratio>::<NRf> is the ratio of the pulse width to the pulse period.

<ripple>::<NRf> is the Ripple is the peak-to-peak ripple on the pulse top.

<ripple_db>::<NRf> is the Ripple measurement expressed in dB.

<droop>::<NRf> is the Droop is the power difference between the beginning and the end of the pulse on time.

<droop_db>::<NRf> is the Droop measurement expressed in dB.

<ovrshoot>::<NRf> is the amount by which the signal exceeds the 100% level on the pulse rising edge.

<ovrshoot_db>::<NRf> is the Overshoot measurement expressed in dB.

<pp_phase>::<NRf> is the phase difference between the selected pulse and the previous pulse in the analysis window.

<pp_freq> ::<NRf> is the difference between the frequency of the current pulse and frequency of the previous pulse.

<rms_phase_err>::<NRf> is the RMS average of the Phase versus Time trace, computed over the Measurement Time.

<rms_freq_err>::<NRf> is the RMS average of the Freq error versusTime trace.

<p2p_phase>::<NRf> is the frequency difference between the current pulse and the first pulse.

<p2p_freq>::<NRf> is the frequency difference between the current pulse and immediate previous pulse.

<pk_phase_err>::<NRf> is the maximum phase error.

<abs_freq>::<NRf> is the absolute pulse frequency measured at a specific point.

<freq_dev>::<NRf> is the difference between the maximum and minimum measured values of the signal frequency during the Measurement Time.

<phase_dev>::<NRf> is the Phase deviation difference between the maximum and minimum Phase values measured during the on time of a pulse.

<dlta_freq>::<NRf> is the difference from the measurement frequency to each pulse frequency.

<imp_resp_amp>::<NRf> is the difference in dB between the levels of the main lobe and highest side lobe.

<imp_resp_time>::<NRf> is the difference in time between the main lobe and highest side lobe .

<time>::<NRf> is the dtime in seconds relative to the time reference point in the first acquisition record in the data set.

Example

FETCH:PULSE[:RESULT]:MEASview1:ADVANCED:RANGe 2048,2060,AVERage might return -3,23.8467659801E-3,23.8566566259E-3,23.8667335361E-3,23.8771531731E-3, 23.8866508007E-3,23.8966420293E-3,23.9066667855E-3,23.9170398563E-3, 23.9367540926E-3,23.9467956126E-3,23.9567346871E-3,23.9670332521E-3, 23.9866096526E-3,2, indicating the average ON value for the pulse number 2048 to 2060.

FETCh:PULSe[:RESult][:MEASview<y>]:ATX? (Query Only)

Returns the average transmitted power in the pulse table.

Conditions

Measurement views: Pulse table

Group

Fetch commands

Syntax

FETCh:PULSe[:RESult][:MEASview<y>]:ATX?

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<first_pulse_num>,<ATX(1)>,< ATX(2)>,...<ATX(n)>

Where

<first_pulse_num>::<NR1> is the first pulse number.

<ATX(n)>::<NRf> is the average transmitted power for the pulse with the number of [first_pulse_num + n - 1] in dBm.

The unit can be changed to watts by the [SENSe]:POWer:UNITs command. This measurement gives valid results only for power units (Watts, dBm, dBuW, dBW).

Examples

FETCH:PULSE:RESULT:MEASview1:ATX? might return 0,-18.57,-18.73,-18.20, -18.53 for Pulse 0 to 3.

FETCh:PULSe[:RESult][:MEASview<y>]:AVERage? (Query Only)

Returns the average on power in the pulse table.

Conditions

Measurement views: Pulse table

Group

Fetch commands

Syntax

FETCh:PULSe[:RESult][:MEASview<y>]:AVERage?

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<first_pulse_num>,< avg(1)>,< avg(2)>,...<avg(n)>`

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

`<avg(n)>::<NRf>` is the average on power for the pulse with the number of [first_pulse_num + n - 1] in dBm.

The unit can be changed by the `[SENSe]:POWer:UNITs` command.

This measurement gives valid results only for power units (Watts, dBm, dBuW, dBW).

Examples

`FETCH:PULSE:RESULT:MEASview1:AVERAGE?` might return `0,-2.354,-2.368,-2.343,-2.358` for Pulse 0 to 3.

FETCh:PULSe[:RESult][:MEASview<y>]:DRODb? (Query Only)

Returns the droop in dB in the pulse table.

Conditions

Measurement views: Pulse table

Group

Fetch commands

Syntax

`FETCh:PULSe[:RESult][:MEASview<y>]:DRODb?`

Related Commands

FETCh:PULSe[:RESult][:MEASview<y>]:DROop? (on page 1314)

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<first_pulse_num>,<droop(1)>,<droop(2)>,...<droop(n)>`

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

`<droop(n)>::<NRf>` is the droopdB for the pulse number [first_pulse_num + n - 1] in percent (%) of watts or volts.

Examples

`FETCH:PULSE:RESULT:MEASview1:DRODB?` might return `1,-0.4,0.32,0.2,-0.1` for Pulse 1 to 4.

FETCh:PULSe[:RESult][:MEASview<y>]:DROop? (Query Only)

Returns the droop in the pulse table.

Conditions

Measurement views: Pulse table

Group

Fetch commands

Syntax

`FETCh:PULSe[:RESult][:MEASview<y>]:DROop?`

Related Commands

FETCh:PULSe[:RESult][:MEASview<y>]:DRODb? (on page 1314)

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<first_pulse_num>,<droop(1)>,<droop(2)>,...<droop(n)>`

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

`<droop(n)>::<NRf>` is the droop for the pulse number [first_pulse_num + n - 1] in percent (%) of watts or volts.

Examples

`FETCH:PULSE:RESULT:MEASview1:DROOP?` might return `0,-270.9E-3,-193.0E-3,-242.7E-3,-177.5E-3` for Pulse 0 to 3.

FETCh:PULSe[:RESult][:MEASview<y>]:DUTPct? (Query Only)

Returns the duty factor (%) in the pulse table.

Conditions

Measurement views: Pulse table

Group

Fetch commands

Syntax

`FETCh:PULSe[:RESult][:MEASview<y>]:DUTPct?`

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<first_pulse_num>,<duty_pct(1)>,<duty_pct(2)>,...<duty_pct(n)>`

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

`<duty_pct(n)>::<NRf>` is the duty factor for the pulse with the number of [first_pulse_num + n - 1] in percent (%).

Examples

FETCH:PULSE:RESULT:MEASview1:DUTPCT? might return 0,28.94,28.96,29.00,29.01 for Pulse 0 to 3.

FETCh:PULSe[:RESult][:MEASview<y>]:DUTRatio? (Query Only)

Returns the duty factor (ratio) in the pulse table.

Conditions

Measurement views: Pulse table

Group

Fetch commands

Syntax

FETCh:PULSe[:RESult][:MEASview<y>]:DUTRatio?

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<first_pulse_num>,<duty_ratio(1)>,<duty_ratio(2)>,...<duty_ratio(n)>

Where

<first_pulse_num>::<NR1> is the first pulse number.

<duty_ratio(n)>::<NRf> is the duty factor for the pulse with the number of [first_pulse_num + n - 1] (no unit).

Examples

FETCH:PULSE:RESULT:MEASview1:DUTRATIO? might return 0,289.4E-3,289.6E-3, 290.0E-3,290.1E-3 for Pulse 0 to 3.

FETCh:PULSe[:RESult][:MEASview<y>]:FABS? (Query Only)

Returns the computed absolute frequency from the carrier in the pulse table.

Conditions

Measurement views: Pulse table

Group

Fetch commands

Syntax

FETCh:PULSe[:RESult][:MEASview<y>]:FABS?

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<first_pulse_num>,<abs_freq(1)>,<abs_freq(2),. <abs_freq(n)>

Where

<first_pulse_num>::<NR1> is the first pulse number.

<abs_freq(n)>::<NR1> is the absolute frequency for the pulse with the number of [first_pulse_num + n - 1] in Hz.

Examples

FETCH:PULSE[:RESULT]:MEASview1:FABS? might return 1.0000000000, 919.8726400000E+6, 1.1664642560E+9, 1.1866586880E+9, 1.1867939840E+9, 1.1882243840E+9, 933.2595840000E+6, 1.1729175040E+9, 1.1818649600E+9, 1.1868037120E+9, 1.1832893440E+9, 933.4762880000E+6, 1.1805880320E+9, 1.1844794880E+9, 1.1914785280E+9, 1.1929450240E for is the number of the first pulse and the absolute frequency for each pulse.

FETCh:PULSe[:RESult][:MEASview<y>]:FALL? (Query Only)

Returns the fall time in the pulse table.

Conditions

Measurement views: Pulse table

Group

Fetch commands

Syntax

FETCh:PULSe[:RESult][:MEASview<y>]:FALL?

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<first_pulse_num>,<fall(1)>,<fall(2)>,...<fall(n)>

Where

<first_pulse_num>::<NR1> is the first pulse number.

<fall(n)>::<NRf> is the fall time for the pulse with the number of [first_pulse_num + n - 1] in seconds.

Examples

FETCH:PULSE:RESULT:MEASview1:FALL? might return 0,110.3E-9,90.45E-9, 95.03E-9,111.9E-9 for Pulse 0 to 3.

FETCh:PULSe[:RESult][:MEASview<y>]:FDELta? (Query Only)

Returns the computed delta frequency from the carrier in the pulse table.

Conditions

Measurement views: Pulse table

Group

Fetch commands

Syntax

FETCh:PULSe[:RESult][:MEASview<y>]:FDELta?

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<first_pulse_num>,<freq_delta(1)>,<freq_delta(2)>,...<freq_delta(n)>

Where

<first_pulse_num>::<NR1> is the first pulse number.

<freq_delta(n)>::<NRf> is the frequency delta for the pulse with the number of [first_pulse_num + n - 1] in Hz.

Examples

FETCH:PULSE:RESULT:MEASview1:FDELTA? might return 1.0000000000,163.6692962646,82.8298492432,21.8995475769,89.2963562012,4 for the first pulse and each pulse found.

FETCh:PULSe[:RESult][:MEASview<y>]:FRDeviation? (Query Only)

Returns the frequency deviation in the pulse table.

Conditions

Measurement views: Pulse table

Group

Fetch commands

Syntax

FETCh:PULSe[:RESult][:MEASview<y>]:FRDeviation?

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<first_pulse_num>,<freq_dev(1)>,<freq_dev(2)>,...<freq_dev(n)>

Where

<first_pulse_num>::<NR1> is the first pulse number.

<freq_dev(n)>::<NRf> is the frequency deviation for the pulse with the number of [first_pulse_num + n - 1] in Hz.

Examples

FETCH:PULSE:RESULT:MEASview1:FRDEVIATION? might return 1,740.6E+3, 736.5E+3,718.3E+3,672.2E+3 for Pulse 1 to 4.

FETCh:PULSe[:RESult][:MEASview<y>]:IRAMplitude? (Query Only)

Returns the pulse impulse response amplitude measurement in the pulse table.

Conditions

Measurement views: Pulse table

Group

Fetch commands

Syntax

`FETCh:PULSe[:RESult][:MEASview<y>]:IRAMplitude?`

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<first_pulse_num>,<amplitude(1)>,<amplitude(2)>,...<amplitude(n)>`

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

`<amplitude(n)>::<NRf>` is the sidelobe level, in dB, from the mainlobe for the pulse with the number of [first_pulse_num + n - 1].

Examples

`FETCH:PULSE:RESULT:MEASview1:IRAMPLITUDE?` might return `1.0000000000,-45.3,-47.5,-50.2` for Pulse 1 through 3.

FETCh:PULSe[:RESult][:MEASview<y>]:IRTime? (Query Only)

Returns the impulse response time in the pulse table.

Conditions

Measurement views: Pulse table

Group

Fetch commands

Syntax

`FETCh:PULSe[:RESult][:MEASview<y>]:IRTime?`

Returns

`<first_pulse_num>, <time(1>), <time(2)>,. <time(n)>`

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

`<time(n)>::<NRf>` is the sidelobe offset time from the mainlobe for the pulse with the number of [first_pulse_num + n - 1] in seconds.

Examples

`FETCH:PULSE:RESULT:MEASview1:IRTIME?` might return `1.0000000000,806.6708687693E-9,-846.6595318168E-9,-873.3477443457E-9` for Pulse 1 through 3.

FETCh:PULSe[:RESult][:MEASview<y>]:MFReqerror? (Query Only)

Returns the maximum frequency error in the pulse table.

Conditions

Measurement views: Pulse table

Group

Fetch commands

Syntax

`FETCh:PULSe[:RESult][:MEASview<y>]:MFReqerror?`

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<first_pulse_num>,<max_freq_err(1)>,<max_freq_err(2)>,...<max_freq_err(n)>`

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

`<max_freq_err(n)>::<NRf>` is the maximum frequency error for the pulse with the number of [first_pulse_num + n - 1] in Hz.

Examples

`FETCH:PULSE:RESULT:MEASview1:MFREQERROR?` might return `1,597.5E+3,675.8E+3,642.8E+3,598.2E+3` for Pulse 1 to 4.

FETCh:PULSe[:RESult][:MEASview<y>]:MPHerror? (Query Only)

Returns the maximum phase error in the pulse table.

Conditions

Measurement views: Pulse table

Group

Fetch commands

Syntax

`FETCh:PULSe[:RESult][:MEASview<y>]:MPHerror?`

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<first_pulse_num>,<max_phase_err(1)>,<max_phase_err(2)>,...<max_phase_err(n)>`

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

<max_phase_err(n)>::<NRf> is the maximum phase error for the pulse with the number of [first_pulse_num + n - 1] in degrees.

Examples

FETCH:PULSE:RESULT:MEASview1:MPHERROR? might return 1,-9.221,-8.413, -11.853,-10.258 for Pulse 1 to 4.

FETCh:PULSe[:RESult][:MEASview<y>]:OVEDb? (Query Only)

Returns the overshoot in dB in the pulse table.

Conditions

Measurement views: Pulse table

Group

Fetch commands

Syntax

FETCh:PULSe[:RESult][:MEASview<y>]:OVEDb?

Related Commands

FETCh:PULSe[:RESult][:MEASview<y>]:OVERshoot? (on page 1321)

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<first_pulse_num>,<overshoot(1)>,<overshoot(2)>,...<overshoot(n)>

Where

<first_pulse_num>::<NR1> is the first pulse number.

<overshoot(n)>::<NRf> is the overshoot for the pulse with the number of [first_pulse_num + n - 1] in dB.

Examples

FETCH:PULSE:RESULT:MEASview1:OVERDB? might return 1,1.2,2.3,1.0,0.2 for Pulse 1 to 4.

FETCh:PULSe[:RESult][:MEASview<y>]:OVERshoot? (Query Only)

Returns the overshoot in the pulse table.

Conditions

Measurement views: Pulse table

Group

Fetch commands

Syntax

FETCh:PULSe[:RESult][:MEASview<y>]:OVERshoot?

Related Commands

FETCh:PULSe[:RESult][:MEASview<y>]:OVEDb? (on page 1321)

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<first_pulse_num>,<overshoot(1)>,<overshoot(2)>,...<overshoot(n)>`

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

`<overshoot(n)>::<NRf>` is the overshoot for the pulse with the number of [first_pulse_num + n - 1] in percent (%) of watts or volts.

Examples

`FETCH:PULSE:RESULT:MEASview1:OVERSHOOT?` might return `1,1.2,2.3,1.0,0.2` for Pulse 1 to 4.

FETCh:PULSe[:RESult][:MEASview<y>]:PHDeviation? (Query Only)

Returns the phase deviation in the pulse table.

Conditions

Measurement views: Pulse table

Group

Fetch commands

Syntax

`FETCh:PULSe[:RESult][:MEASview<y>]:PHDeviation?`

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<first_pulse_num>,<phase_dev(1)>,<phase_dev(2)>,...<phase_dev(n)>`

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

`<phase_dev(n)>::<NRf>` is the phase deviation for the pulse with the number of [first_pulse_num + n - 1] in degrees.

Examples

`FETCH:PULSE:RESULT:MEASview1:PHDEVIATION?` might return `1,11.658,9.640,10.509,8.272` for Pulse 1 to 4.

FETCh:PULSe[:RESult][:MEASview<y>]:PPFD? (Query Only)

Returns the computed pulse to pulse frequency difference from the carrier in the pulse table.

Conditions

Measurement views: Pulse table

Group

Fetch commands

Syntax

`FETCh:PULSe[:RESult][:MEASview<y>]:PPFD?`

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<first_pulse_num>,<ppfd(1)>,<ppfd(2),. <ppfd(n)>`

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

`<ppfd(n)>::<NRf>` is the pulse to pulse frequency difference with the number of [first_pulse_num + n - 1] in Hz.

Examples

`FETCH:PULSE[:RESULT]:MEASview1:PPFD?` might return `1.0000000000,"-", 246.5916160000E+6, 20.1943520000E+6, 135.2960000000E+3, 1.4304160000E+6, -254.9647520000E+6, 239.6579520000E+6, 8.9474560000E+6, 4.9386560000E+6, -3.5143840000E+6, -249.8130240000E+6, 247.1117760000E+6, 3.8915200000E+6, 6.9989120000E+6, 1.4665920000E+6` is the number of the first pulse and the pulse frequency difference for each pulse.

FETCh:PULSe[:RESult][:MEASview<y>]:PPFRequency? (Query Only)

Returns the pulse-pulse carrier frequency in the pulse table.

Conditions

Measurement views: Pulse table

Group

Fetch commands

Syntax

`FETCh:PULSe[:RESult][:MEASview<y>]:PPFRequency?`

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<first_pulse_num>,<pp_freq(1)>,<pp_freq(2)>,...<pp_freq(n)>`

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

`<pp_freq(n)>::<NRf>` is the pulse-pulse carrier frequency for the pulse with the number of [first_pulse_num + n - 1] in Hz.

Examples

`FETCH:PULSE:RESULT:MEASview1:PPFREQUENCY?` might return `0,0.000,1.258E+3,-3.121E+3,1.862E+3` for Pulse 0 to 3.

FETCh:PULSe[:RESult][:MEASview<y>]:PPOWer? (Query Only)

Returns the peak power in the pulse table.

Conditions

Measurement views: Pulse table

Group

Fetch commands

Syntax

`FETCh:PULSe[:RESult][:MEASview<y>]:PPOWer?`

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<first_pulse_num>,<pk_power(1)>,<pk_power(2)>,...<pk_power(n)>`

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

`<pk_power(n)>::<NRf>` is the peak power for the pulse with the number of [first_pulse_num + n - 1] in dBm.

The unit can be changed by the `[SENSe]:POWer:UNITs` command.

This measurement gives valid results only for power units(Watts, dBm, dBuW, dBW).

Examples

`FETCH:PULSE:RESULT:MEASview1:PPOWER?` might return `0,-2.26,-2.27,-2.23,-2.25` for Pulse 0 to 3.

FETCh:PULSe[:RESult][:MEASview<y>]:PPPD? (Query Only)

Returns the computed pulse to pulse phase difference from the carrier in the pulse table.

Conditions

Measurement views: Pulse table

Group

Fetch commands

Syntax

`FETCh:PULSe[:RESult][:MEASview<y>]:PPPD?`

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<first_pulse_num>,<pppd(1)>,<pppd(2),. <pppd(n)>

Where

<first_pulse_num>::<NR1> is the first pulse number.

<pppd(n)>::<NRf> is the pulse to pulse phase difference with the number of [first_pulse_num + n - 1] in Hz.

Examples

FETCH:PULSE[:RESULT]:MEASview1:PPPD? might return 1.0000000000,"-", 96.2469482422, 24.9118175507, 5.8206896782, 16.7037734985, -151.8063964844, 100.7584228516,36.6761512756, -3.8614616394, 12.9028282166, -150.9820556641, 130.5854492188, 9.1259155273, -7.4177660942, 6.0613451004, -142.4224853516 is the number of the first pulse and the pulse to pulse phase difference for each pulse.

FETCh:PULSe[:RESult][:MEASview<y>]:PPPHase? (Query Only)

Returns the pulse-pulse carrier phase in the pulse table.

Conditions

Measurement views: Pulse table

Group

Fetch commands

Syntax

FETCh:PULSe[:RESult][:MEASview<y>]:PPPHase?

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<first_pulse_num>,<pp_phase(1)>,<pp_phase(2)>,...<pp_phase(n)>

Where

<first_pulse_num>::<NR1> is the first pulse number.

<pp_phase(n)>::<NRf> is the pulse-pulse carrier phase for the pulse with the number of [first_pulse_num + n - 1] in degrees.

Examples

FETCH:PULSE:RESULT:MEASview1:PPPHASE? might return 0,0.000,21.66,46.76,57.56 for Pulse 0 to 3.

FETCh:PULSe[:RESult]:MEASview<y>:RINTerval? (Query Only)

Returns the repetition interval in the pulse table.

Conditions

Measurement views: Pulse table

Group

Fetch commands

Syntax

`FETCh:PULSe[:RESult]:MEASview<y>:RINTerval?`

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<first_pulse_num>,<rep_int(1)>,<rep_int(2)>,...<rep_int(n)>`

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

`<rep_int(n)>::<NRf>` is the repetition interval for the pulse with the number of [first_pulse_num + n - 1] in seconds.

Examples

`FETCH:PULSE:RESULT:MEASview1:RINTERVAL?` might return `0,16.03E-6,16.08E-6,16.07E-6,16.02E-6` for Pulse 0 to 3.

FETCh:PULSe[:RESult][:MEASview<y>]:RIPDb? (Query Only)

Returns the ripple in dB in the pulse table.

Conditions

Measurement views: Pulse table

Group

Fetch commands

Syntax

`FETCh:PULSe[:RESult][:MEASview<y>]:RIPDb?`

Related Commands

FETCh:PULSe[:RESult][:MEASview<y>]:RIPPle? (on page 1327)

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<first_pulse_num>,<ripple(1)>,<ripple(2)>,...<ripple(n)>`

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

`<ripple(n)>::<NRf>` is the ripple for the pulse with the number of [first_pulse_num + n – 1] in dB.

Examples

`FETCH:PULSE:RESULT:MEASview1:RIPDB?` might return 1,0.4,0.32,0.2,0.1 for Pulse 1 to 4.

FETCh:PULSe[:RESult][:MEASview<y>]:RIPPle? (Query Only)

Returns the ripple in the pulse table.

Conditions

Measurement views: Pulse table

Group

Fetch commands

Syntax

`FETCh:PULSe[:RESult][:MEASview<y>]:RIPPle?`

Related Commands

FETCh:PULSe[:RESult][:MEASview<y>]:RIPDb? (on page 1326)

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<first_pulse_num>,<ripple(1)>,<ripple(2)>,...<ripple(n)>`

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

`<ripple(n)>::<NRf>` is the volts or watts ripple for the pulse with the number of [first_pulse_num + n - 1] in percent (%).

Examples

`FETCH:PULSE:RESULT:MEASview1:RIPPLE?` might return `0,106.5E-3,177.6E-3, 148.3E-3,148.5E-3` for Pulse 0 to 3.

FETCh:PULSe[:RESult][:MEASview<y>]:RISE? (Query Only)

Returns the rise time in the pulse table.

Conditions

Measurement views: Pulse table

Group

Fetch commands

Syntax

`FETCh:PULSe[:RESult][:MEASview<y>]:RISE?`

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<first_pulse_num>,<rise(1)>,<rise(2)>,...<rise(n)>

Where

<first_pulse_num>::<NR1> is the first pulse number.

<rise(n)>::<NRf> is the rise time for the pulse with the number of [first_pulse_num + n - 1] in seconds.

Examples

FETCH:PULSE:RESULT:MEASview1:RISE? might return 0,92.94E-9,115.9E-9, 115.1E-9,97.45E-9 for Pulse 0 to 3.

FETCh:PULSe[:RESult][:MEASview<y>]:RMSFreqerror? (Query Only)

Returns the RMS frequency error in the pulse table.

Conditions

Measurement views: Pulse table

Group

Fetch commands

Syntax

FETCh:PULSe[:RESult][:MEASview<y>]:RMSFreqerror?

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<first_pulse_num>,<RMS_freq_err(1)>,<RMS_freq_err(2)>,...<RMS_freq_err(n)>

Where

<first_pulse_num>::<NR1> is the first pulse number.

<RMS_freq_err(n)>::<NRf> is the RMS frequency error for the pulse with the number of [first_pulse_num + n - 1] in Hz.

Examples

FETCH:PULSE:RESULT:MEASview1:RMSFREQERROR? might return 1,51.54E+3, 69.20E+3,64.21E+3,51.02E+3 for Pulse 1 to 4.

FETCh:PULSe[:RESult][:MEASview<y>]:RMSPherror? (Query Only)

Returns the RMS phase error in the pulse table.

Conditions

Measurement views: Pulse table

Group

Fetch commands

Syntax

`FETCh:PULSe[:RESult][:MEASview<y>]:RMSPherror?`

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<first_pulse_num>,<RMS_phase_err(1)>,<RMS_phase_err(2)>,...<RMS_phase_err(n)>`

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

`<RMS_phase_err(n)>::<NRf>` is the RMS phase error for the pulse with the number of [first_pulse_num + n - 1] in degrees.

Examples

`FETCH:PULSE:RESULT:MEASview1:RMSPHERROR?` might return `1,908.4E-3,752.8E-3,981.7E-3,886.4E-3` for Pulse 1 to 4.

FETCh:PULSe[:RESult][:MEASview<y>]:RRATe? (Query Only)

Returns the repetition rate in the pulse table.

Conditions

Measurement views: Pulse table

Group

Fetch commands

Syntax

`FETCh:PULSe[:RESult][:MEASview<y>]:RRATe?`

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<first_pulse_num>,<rep_rate(1)>,<rep_rate(2)>,...<rep_rate(n)>`

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

`<rep_rate(n)>::<NRf>` is the repetition rate for the pulse with the number of [first_pulse_num + n - 1] in Hz.

Examples

`FETCH:PULSE:RESULT:MEASview1:RRATE?` might return `0,62.50E+3,62.52E+3, 62.51E+3,62.49E+3` for Pulse 0 to 3.

FETCh:PULSe[:RESult][:MEASview<y>]:TIME? (Query Only)

Returns the time in the pulse table.

Conditions

Measurement views: Pulse table

Group

Fetch commands

Syntax

```
FETCh:PULSe[:RESult][:MEASview<y>]:TIME?
```

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

```
<first_pulse_num>,<time(1)>,<time(2)>,...<time(n)>
```

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

`<time(n)>::<NRf>` is the time for the pulse with the number of [first_pulse_num + n - 1] in seconds.

Examples

`FETCH:PULSE:RESULT:MEASview1:TIME?` might return `1,7.937E-3,8.436E-3, 6.504E-3,9.876E-3` for Pulse 1 to 4.

FETCh:PULSe[:RESult][:MEASview<y>]:WIDTh? (Query Only)

Returns the pulse width in the pulse table.

Conditions

Measurement views: Pulse table

Group

Fetch commands

Syntax

```
FETCh:PULSe[:RESult][:MEASview<y>]:WIDTh?
```

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

```
<first_pulse_num>,<width(1)>,<width(2)>,...<width(n)>
```

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

`<width(n)>::<NRf>` is the pulse width for the pulse with the number of [first_pulse_num + n - 1] in seconds.

Examples

FETCH:PULSE:RESULT:MEASview1:WIDTH? might return 0,4.630E-6,4.632E-6, 4.639E-6,4.642E-6 for Pulse 0 to 3.

FETCH:PULSe:STATistics[:MEASview<y>]:ADVanced (Query Only)

Returns the Pulse Count, Max, Max at Pulse number, Min, Min at Pulse number, Peak to Peak, Average and Standard Deviation values for the chosen measurement in Pulse Statistics display.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Fetch commands

Syntax

FETCH:PULSe:STATistics[:MEASview<y>]:ADVanced? <arg1>,<arg2>

Arguments

<arg1>= {AVERage | PPOWer | ATX | WIDTh | RISE | FALL | RINTerval | RRATe | DUTPct | DUTRatio |RIPPle | RIPDb | DROop | DRODb | OVERshoot | OVEDb | PPPHase | PPFRequency | RMSFreqerror | MFReqerror | RMSPherror | MPHerror | FRDeviation | FDELta | PHDeviation | IRAMplitude|TIME | IRTime | PPPD| PPFD | FABS}

<arg2>={TPCOunt|MAX|MIN|PK2PK|AVG|SD|ALL}

Returns

<puls_cnt> | <max> | <min> | <pk_pk> | <avg> | <std_dev> | <all> For one of the following selected measurements:

<avg_on> | <pk_pwr> | <avg_tx> | <width> | <rise> | <fall> | <rep_int> | <rep_rate> | <duty_factor_pcnt> | <duty_fact_ratio> | <ripple> | <ripple_db> | <droop> | <droop_db> | <ovrshoot> | <ovrshoot_db> | <pp_phase> | <pp_freq> | <rms_phase_err> | <rms_freq_err> | <p2p_phase> | <p2p_freq> | <pk_phase_err> | <pk_freq_err> | <abs_freq> | <freq_dev> | <phase_dev> | <dlta_freq> | <imp_resp_amp> | <imp_resp_time>|<time> for the given pulse number

where

<avg_on>::<NRf> is the average power transmitted during pulse on.

<pk_pwr>::<NRf> is the Maximum power during pulse on

<avg_tx>:: <NRf> is the average power transmitted

<width> is the time from the rising edge to the falling edge at the –3 dB/–6 dB level (50%) of the user selected 100% level

<rise> is the time required for a signal to rise from 10% to 90% (or 20% to 80%) of the user selected 100% level

<fall> ::<NRf> is the time required for a signal to fall from 90% to 10% (or 80% to 20%) of the user selected 100% level

<rep_int>::<NRf> is the time from a pulse rising edge to the next pulse rising edge

<rep_rate> ::<NRf> is the inverse of repetition interval

<duty_factor_pcnt>::<NRf> is the ratio of the width to the pulse period, expressed as a percentage

<duty_fact_ratio>::<NRf> is the ratio of the pulse width to the pulse period

<ripple>::<NRf> is the Ripple is the peak-to-peak ripple on the pulse top

<ripple_db>::<NRf> is the Ripple measurement expressed in dB

<droop>::<NRf> is the Droop is the power difference between the beginning and the end of the pulse On time

<droop_db>::<NRf> is the Droop measurement expressed in dB

<ovrshoot>::<NRf> is the amount by which the signal exceeds the 100% level on the pulse rising edge

<ovrshoot_db>::<NRf> is the Overshoot measurement expressed in dB

<pp_phase>::<NRf> is the phase difference between the selected pulse and the previous pulse in the analysis window

<pp_freq> ::<NRf> is the difference between the frequency of the current pulse and frequency of the previous pulse

<rms_phase_err>::<NRf> is the RMS average of the Phase vs Time trace, computed over the Measurement Time

<rms_freq_err>::<NRf> is the RMS average of the Freq Error vs.Time trace

<p2p_phase>::<NRf> is the frequency difference between the current pulse and the first pulse

<p2p_freq>::<NRf> is the frequency difference between the current pulse and immediate previous pulse

<pk_phase_err>::<NRf> is the maximum phase error

<abs_freq>::<NRf> is the absolute pulse frequency measured at a specific point

<freq_dev>::<NRf> is the difference between the maximum and minimum measured values of the signal frequency during the Measurement Time

<phase_dev>::<NRf> is the Phase Deviation difference between the maximum and minimum Phase values measured during the On time of a pulse

<dlta_freq>::<NRf> is the difference from the measurement frequency to each pulse frequency

<imp_resp_amp>::<NRf> is the difference in dB between the levels of the main lobe and highest side lobe

<imp_resp_time>::<NRf> is the difference in time between the main lobe and highest side lobe

<time>::<NRf> is the dtime in seconds relative to the time reference point in the first acquisition record in the data set

Examples

FETCH:PULSE:STATISTICS:MEASview1:ADVANCED RISE, MAX might return 7.9786177665E-9,"-", indicating the maximum rise value is 7.9786177665E-9.

FETCh:PULSe:STATistics[:MEASview<y>]:ATX (Query Only)

Returns the average transmitted power in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT is set to TRENd.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Fetch commands

Syntax

FETCh:PULSe:STATistics[:MEASview<y>]:ATX?

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<ATX_avg>,<ATX_min>,<ATX_max>

Where

<ATX_avg>::<NRf> is the average of the average transmitted power.

<ATX_min>::<NRf> is the minimum of the average transmitted power.

<ATX_max>::<NRf> is the maximum of the average transmitted power. Unit: dBm.

The unit can be changed to Watts by the [SENSe]:POWer:UNITs command. Volt is invalid in the average transmitted power measurement.

Examples

FETCH:PULSE:STATISTICS:MEASVIEW2:ATX? might return -18.51,-18.74,-18.12 for the average transmitted power in the pulse statistics.

FETCh:PULSe:STATistics[:MEASview<y>]:AVERage (Query Only)

Returns the average on power in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT is set to TRENd.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Fetch commands

Syntax

FETCh:PULSe:STATistics[:MEASview<y>]:AVERage?

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<avg_avg>,<avg_min>,<avg_max>

Where

<avg_avg>::<NRf> is the average of the average on power.

<avg_min>::<NRf> is the minimum of the average on power.

<avg_max>::<NRf> is the maximum of the average on power. Unit: dBm.

The unit can be changed to watts by the [SENSe]:POWer:UNITs command.

Examples

FETCH:PULSE:STATISTICS:MEASVIEW2:AVERage? might return -2.35,-2.36,-2.34 for the average on power in the pulse statistics.

FETCh:PULSe:STATistics[:MEASview<y>]:DRODb (Query Only)

Returns the droop in dB in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT is set to TRENd.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Fetch commands

Syntax

`FETCh:PULSe:STATistics[:MEASview<y>]:DRODb?`

Related Commands

FETCh:PULSe:STATistics[:MEASview<y>]:DROop? (on page 1334)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<droop_avg>,<droop_min>,<droop_max>`

Where

`<droop_avg>::<NRf>` is the average droop.

`<droop_min>::<NRf>` is the minimum droop.

`<droop_max>::<NRf>` is the maximum droop. Unit: in dB.

Examples

`FETCH:PULSE:STATISTICS:MEASVIEW2:DRODb?` might return `22.67E-3,-613.5E-3, 633.8E-3` for the droop in the pulse statistics.

FETCh:PULSe:STATistics[:MEASview<y>]:DROop (Query Only)

Returns the droop in the pulse statistics. This command is valid when `DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT` is set to `TRENd`.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Fetch commands

Syntax

`FETCh:PULSe:STATistics[:MEASview<y>]:DROop?`

Related Commands

FETCh:PULSe:STATistics[:MEASview<y>]:DRODb? (on page 1333)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<droop_avg>,<droop_min>,<droop_max>`

Where

<droop_avg>::<NRf> is the average droop.

<droop_min>::<NRf> is the minimum droop.

<droop_max>::<NRf> is the maximum droop. Unit: Percent (%) by watts.

Examples

FETCH:PULSE:STATISTICS:MEASVIEW2:DROop? might return 22.67E-3,-613.5E-3, 633.8E-3 for the droop in the pulse statistics.

FETCh:PULSe:STATistics[:MEASview<y>]:DUTPct (Query Only)

Returns the duty factor (%) in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT is set to TRENd.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Fetch commands

Syntax

FETCh:PULSe:STATistics[:MEASview<y>]:DUTPct?

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<duty_pct_avg>,<duty_pct_min)>,<duty_pct_max>

Where

<duty_pct_avg>::<NRf> is the average duty factor.

<duty_pct_min>::<NRf> is the minimum duty factor.

<duty_pct_max>::<NRf> is the maximum duty factor. Unit: Percent (%).

Examples

FETCH:PULSE:STATISTICS:MEASVIEW2:DUTPct? might return 2.437,2.310,2.657 for the duty factor (%) in the pulse statistics.

FETCh:PULSe:STATistics[:MEASview<y>]:DUTRatio (Query Only)

Returns the duty factor (ratio) in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT is set to TRENd.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Fetch commands

Syntax

FETCh:PULSe:STATistics[:MEASview<y>]:DUTRatio?

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<duty_ratio_avg>,<duty_ratio_min>,<duty_ratio_max)>

Where

<duty_ratio_avg>::<NRf> is the average duty factor.

<duty_ratio_min>::<NRf> is the minimum duty factor.

<duty_ratio_max>::<NRf> is the maximum duty factor. Unit: None.

Examples

FETCH:PULSE:STATISTICS:MEASVIEW2:DUTRATIO? might return 24.37E-3,23.11E-3, 26.57E-3 for the duty factor (ratio) in the pulse statistics.

FETCh:PULSe:STATistics[:MEASview<y>]:FABS (Query Only)

Returns the absolute frequency in the Pulse Statistics display.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Fetch commands

Syntax

FETCh:PULSe:STATistics[:MEASview<y>]:FABS?

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf>

Examples

FETCH:PULSE:STATISTICS:MEASVIEW2:FABS? might return 99.9846400000E+6, 99.9846400000E+6, 99.9846400000E+6, indicating the absolute frequency value.

FETCh:PULSe:STATistics[:MEASview<y>]:FALL (Query Only)

Returns the fall time in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT is set to TRENd.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Fetch commands

Syntax

`FETCh:PULSe:STATistics[:MEASview<y>]:FALL?`

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<fall_avg>,<fall_min>,<fall_max>`

Where

`<fall_avg>::<NRf>` is the average fall time.

`<fall_min>::<NRf>` is the minimum fall time.

`<fall_max>::<NRf>` is the maximum fall time. Unit: Seconds.

Examples

`FETCH:PULSE:STATISTICS:MEASVIEW2:FALL?` might return `70.27E-9,69.62E-9, 71.27E-9` for the fall time in the pulse statistics.

FETCh:PULSe:STATistics[:MEASview<y>]:FDELta (Query Only)

Return the delta frequency in the pulse statistics.

Group

Fetch commands

Syntax

`FETCh:PULSe:STATistics[:MEASview<y>]:FDELta?`

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<freq_delta_avg>,<freq_delta_min>,<freq_delta_max>`

Where

`<freq_delta_avg>::<NRf>` is the average frequency deviation.

`<freq_delta_min>::<NRf>` is the minimum frequency deviation.

`<freq_delta_max>::<NRf>` is the maximum frequency deviation. Unit: Hz.

Examples

`FETCh:PULSe:STATistics:MEASview1:FDELta?` might return `77.9785690308,-110.1940994263,287.8957519531` for the frequency delta in the pulse statistics.

FETCh:PULSe:STATistics[:MEASview<y>]:FRDeviation (Query Only)

Returns the frequency deviation in the pulse statistics. This command is valid when `DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT` is set to `TRENd`.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Fetch commands

Syntax

```
FETCh:PULSe:STATistics[:MEASview<y>]:FRDeviation?
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<freq_dev_avg>,<freq_dev_min>,<freq_dev_max>`

Where

`<freq_dev_avg>::<NRf>` is the average frequency deviation.

`<freq_dev_min>::<NRf>` is the minimum frequency deviation.

`<freq_dev_max>::<NRf>` is the maximum frequency deviation. Unit: Hz.

Examples

`FETCH:PULSE:STATISTICS:MEASVIEW2:FRDEVIATION?` might return `754.1E+3, 660.5E+3,835.2E+3` for the frequency deviation in the pulse statistics.

FETCh:PULSe:STATistics[:MEASview<y>]:IRAMplitude (Query Only)

Returns the Pulse Impulse Response Amplitude measurement in the pulse statistics.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Fetch commands

Syntax

```
FETCh:PULSe:STATistics[:MEASview<y>]:IRAMplitude?
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<amp_avg>,<amp_min>,<amp_max>`

Where

<amp_avg>::<NRf> is the average of the impulse response amplitude.

<amp_min>::<NRf> is the minimum of the impulse response amplitude.

<amp_max>::<NRf> is the maximum of impulse response amplitude. Unit: dB.

Examples

FETCH:PULSE:STATISTICS:MEASVIEW2:IRAMPLITUDE? might return -45.3,-47.1,-43.2 for the impulse response amplitude in the pulse statistics.

FETCh:PULSe:STATistics[:MEASview<y>]:IRTime (Query Only)

Returns the impulse response time in the pulse statistics.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Fetch commands

Syntax

FETCh:PULSe:STATistics[:MEASview<y>]:IRTime?

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<avg_avg>,<avg_min>,<avg_max>

Where

<avg_avg>::<NRf> is the average of the impulse response time.

<avg_min>::<NRf> is the minimum of the impulse response time.

<avg_max>::<NRf> is the maximum of impulse response time. Unit: seconds.

Examples

FETCH:PULSE:STATISTICS:MEASVIEW2:IRTime? might return -2.35,-2.36,-2.34 for the impulse response time in the pulse statistics.

FETCH:PULSe:STATistics[:MEASview<y>]:MAX:PULSe:NUMber (Query Only)

Returns the maximum value at a certain pulse number in the pulse statistics display.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Fetch commands

Syntax

FETCH:PULSe:STATistics[:MEASview<y>]:MAX:PULSe:NUMber?

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<max_pulse_no>

Where

<max_pulse_no>::<NRf> is the maximum value at a pulse number.

Examples

FETCH:PULSE:STATISTICS:MEASview2:MAX:PULSE:NUMBER? might return 15 for the maximum pulse number is 15.

FETCh:PULSe:STATistics[:MEASview<y>]:MFReqerror (Query Only)

Returns the maximum frequency error in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT is set to TRENd.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Fetch commands

Syntax

FETCh:PULSe:STATistics[:MEASview<y>]:MFReqerror?

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<max_freq_err_avg>,<max_freq_err_min>,<max_freq_err_max>

Where

<max_freq_err_avg>::<NRf> is the average of maximum frequency error.

<max_freq_err_min>::<NRf> is the minimum of maximum frequency error.

<max_freq_err_max>::<NRf> is the maximum of maximum frequency error. Unit: Hz.

Examples

FETCH:PULSE:STATISTICS:MEASVIEW2:MFReqerror? might return 645.0E+3,555.6E+3,738.8E+3 for the maximum frequency error in the pulse statistics.

FETCH:PULSe:STATistics[:MEASview<y>]:MIN:PULSe:NUMber (Query Only)

Returns the minimum value at a certain pulse number in the pulse statistics.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number.

Group

Fetch commands

Syntax

`FETCH:PULSe:STATistics[:MEASview<y>]:MIN:PULSe:NUMber?`

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<min_pulse_no>`

Where

`<min_pulse_no>::<NRf>` is the minimum value at a pulse number.

Examples

`FETCH:PULSE:STATISTICS:MIN:PULSE:NUMBER?` might return `1` for the minimum pulse number in the pulse statistics.

FETCh:PULSe:STATistics[:MEASview<y>]:MPHerror (Query Only)

Returns the maximum phase error in the pulse statistics. This command is valid when `DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT` is set to `TRENd`.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Fetch commands

Syntax

`FETCh:PULSe:STATistics[:MEASview<y>]:MPHerror?`

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<max_phase_err_avg>,<max_phase_err_min>,<max_phase_err_max>`

Where

`<max_phase_err_avg>::<NRf>` is the average of maximum phase error.

`<max_phase_err_min>::<NRf>` is the minimum of maximum phase error.

`<max_phase_err_max>::<NRf>` is the maximum of maximum phase error. Unit: Degrees.

Examples

FETCH:PULSE:STATISTICS:MEASVIEW2:MPHerror? might return -11.47,-17.18, -7.61 for the maximum phase error in the pulse statistics.

FETCh:PULSe:STATistics[:MEASview<y>]:OVEDb (Query Only)

Returns the overshoot measurement result in dB in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT is set to TRENd or TTRend.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Fetch commands

Syntax

FETCh:PULSe:STATistics[:MEASview<y>]:OVEDb?

Related Commands

FETCh:PULSe:STATistics[:MEASview<y>]:OVERshoot? (on page 1342)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<overshoot_avg>,<overshoot_min>,<overshoot_max>

Where

<overshoot_avg>::<NRf> is the average overshoot.

<overshoot_min>::<NRf> is the minimum overshoot.

<overshoot_max>::<NRf> is the maximum overshoot.

Unit: dB.

Examples

FETCH:PULSE:STATISTICS:MEASVIEW2:OVEDb? might return 0.3,0.1,0.5 for the overshoot measurement result in dB in the pulse statistics.

FETCh:PULSe:STATistics[:MEASview<y>]:OVERshoot (Query Only)

Returns the overshoot in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT is set to TRENd or TTRend.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Fetch commands

Syntax

```
FETCh:PULSe:STATistics[:MEASview<y>]:OVERshoot?
```

Related Commands

FETCh:PULSe:STATistics[:MEASview<y>]:OVEDb? (on page 1342)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<overshoot_avg>,<overshoot_min>,<overshoot_max>

Where

<overshoot_avg>::<NRf> is the average overshoot.

<overshoot_min>::<NRf> is the minimum overshoot.

<overshoot_max>::<NRf> is the maximum overshoot. Unit: Percent (%) by watts or volts.

Examples

FETCH:PULSE:STATISTICS:MEASVIEW2:OVERshoot? might return 0.3,0.1,0.5 for the overshoot in the pulse statistics.

FETCh:PULSe:STATistics[:MEASview<y>]:PHDeviation (Query Only)

Returns the phase deviation in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT is set to TRENd.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Fetch commands

Syntax

```
FETCh:PULSe:STATistics[:MEASview<y>]:PHDeviation?
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<phase_dev_avg>,<phase_dev_min>,<phase_dev_max>

Where

<phase_dev_avg>::<NRf> is the average phase deviation.

<phase_dev_min>::<NRf> is the minimum phase deviation.

<phase_dev_max>::<NRf> is the maximum phase deviation. Unit: Degrees.

Examples

FETCH:PULSE:STATISTICS:MEASVIEW2:PHDeviation? might return 11.678,7.694, 17.374 for the phase deviation in the pulse statistics.

FETCh:PULSe:STATistics[:MEASview<y>]:PPFD (Query Only)

Returns the pulse-pulse frequency difference in the Pulse Statistics display.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Fetch commands

Syntax

FETCh:PULSe:STATistics[:MEASview<y>]:PPFD?

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf>

Examples

FETCH:PULSE:STATISTICS:MEASVIEW2:PPFD? might return 365.4186562500E+3, -267.5911040000E+6, 247.1117760000E+6, indicating the pulse-pulse frequency difference.

FETCh:PULSe:STATistics[:MEASview<y>]:PPFRequency (Query Only)

Returns the pulse-pulse carrier frequency in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT is set to TRENd.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Fetch commands

Syntax

FETCh:PULSe:STATistics[:MEASview<y>]:PPFRequency?

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<pp_freq_avg>,<pp_freq_min)>,<pp_freq_max>

Where

<pp_freq_avg>::<NRf> is the average pulse-pulse carrier frequency.

<pp_freq_min>::<NRf> is the minimum pulse-pulse carrier frequency.

<pp_freq_max>::<NRf> is the maximum pulse-pulse carrier frequency. Unit: Hz.

Examples

FETCH:PULSE:STATISTICS:MEASVIEW2:PPFRequency? might return 1.135E+3,311.3E+3,-262.8E+3 for the pulse-pulse carrier frequency in the pulse statistics.

FETCh:PULSe:STATistics[:MEASview<y>]:PPOWer (Query Only)

Returns the peak power in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT is set to TRENd.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Fetch commands

Syntax

FETCh:PULSe:STATistics[:MEASview<y>]:PPOWer?

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<pk_power_avg>,<pk_power_min>,<pk_power_max>

Where

<pk_power_avg>::<NRf> is the average peak power.

<pk_power_min>::<NRf> is the minimum peak power.

<pk_power_max>::<NRf> is the maximum peak power. Unit: dBm.

The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples

FETCh:PULSe:STATistics:MEASview2:PPOWer? might return -2.273,-2.313,-2.235 for the peak power in the pulse statistics 2.

FETCh:PULSe:STATistics[:MEASview<y>]:PPPD (Query Only)

Returns the overshoot measurement result in dB in the Pulse Statistics. This command is valid when DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT is set to TRENd or TTRend.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Fetch commands

Syntax

FETCh:PULSe:STATistics[:MEASview<y>]:PPPD?

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf>

Examples

FETCh:PULSe:STATistics:MEASview2:PPPD? might return -1.1131216288, -151.8063964844, 130.5854492188, indicating the pulse-pulse phase difference in dB.

FETCh:PULSe:STATistics[:MEASview<y>]:PPPHase (Query Only)

Returns the pulse-pulse carrier phase in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT is set to TRENd or TTRend.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Fetch commands

Syntax

FETCh:PULSe:STATistics[:MEASview<y>]:PPPHase?

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<pp_phase_avg>,<pp_phase_min)>,<pp_phase_max>

Where

<pp_phase_avg>::<NRf> is the average pulse-pulse carrier phase.

<pp_phase_min>::<NRf> is the minimum pulse-pulse carrier phase.

<pp_phase_max>::<NRf> is the maximum pulse-pulse carrier phase. Unit: Degrees.

Examples

FETCh:PULSe:STATistics:MEASview2:PPPHase? might return -9.298E-3,-254.3E-3,311.7E-3 for the pulse-pulse carrier phase in the pulse statistics.

FETCh:PULSe:STATistics[:MEASview<y>]:RINTerval (Query Only)

Returns the repetition interval in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT is set to TRENd.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Fetch commands

Syntax

`FETCh:PULSe:STATistics[:MEASview<y>]:RINTerval?`

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<rep_int_avg>,<rep_int_min>,<rep_int_max>`

Where

`<rep_int_avg>::<NRf>` is the average repetition interval.

`<rep_int_min>::<NRf>` is the minimum repetition interval.

`<rep_int_max>::<NRf>` is the maximum repetition interval.

Unit: Seconds.

Examples

`FETCH:PULSE:STATISTICS:MEASVIEW2:RINTERVAL` might return `240.5E-6,217.9E-6,281.2E-6` for the repetition interval in the pulse statistics 2.

FETCh:PULSe:STATistics[:MEASview<y>]:RIPDb (Query Only)

Returns the ripple in dB in the pulse statistics. This command is valid when `DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT` is set to `TRENd`.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Fetch commands

Syntax

`FETCh:PULSe:STATistics[:MEASview<y>]:RIPDb?`

Related Commands

FETCh:PULSe:STATistics[:MEASview<y>]:RIPPle? (on page 1348)

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<ripple_avg>,<ripple_min>,<ripple_max>`

Where

<ripple_avg>::<NRf> is the average ripple.

<ripple_min>::<NRf> is the minimum ripple.

<ripple_max>::<NRf> is the maximum ripple.

Unit: dB.

Examples

FETCH:PULSE:STATISTICS:MEASVIEW2:RIPDb? might return 160.4E-3,83.78E-3,287.7E-3 for the ripple in dB in the pulse statistics.

FETCh:PULSe:STATistics[:MEASview<y>]:RIPPle (Query Only)

Returns the ripple in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT is set to TRENd.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Fetch commands

Syntax

FETCh:PULSe:STATistics[:MEASview<y>]:RIPPle?

Related Commands

FETCh:PULSe:STATistics[:MEASview<y>]:RIPDb? (on page 1347)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<ripple_avg>,<ripple_min>,<ripple_max>

Where

<ripple_avg>::<NRf> is the average ripple.

<ripple_min>::<NRf> is the minimum ripple.

<ripple_max>::<NRf> is the maximum ripple. Unit: Percent (%) by watts or volts.

Examples

FETCH:PULSE:STATISTICS:MEASVIEW2:RIPPle? might return 160.4E-3,83.78E-3,287.7E-3 for the ripple in the pulse statistics.

FETCh:PULSe:STATistics[:MEASview<y>]:RISE (Query Only)

Returns the rise time in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT is set to TRENd.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Fetch commands

Syntax

`FETCh:PULSe:STATistics[:MEASview<y>]:RISE?`

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<rise_avg>,<rise_min>,<rise_max>`

Where

`<rise_avg>::<NRf>` is the average rise time.

`<rise_min>::<NRf>` is the minimum rise time.

`<rise_max>::<NRf>` is the maximum rise time. Unit: Seconds.

Examples

`FETCH:PULSE:STATISTICS:MEASVIEW2:RISE?` might return `105.4E-9,91.65E-9, 116.2E-9` for the rise time in the pulse statistics.

FETCh:PULSe:STATistics[:MEASview<y>]:RMSFreqerror (Query Only)

Returns the RMS frequency error in the pulse statistics. This command is valid when `DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT` is set to `TRENd`.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Fetch commands

Syntax

`FETCh:PULSe:STATistics[:MEASview<y>]:RMSFreqerror?`

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<RMS_freq_err_avg>,<RMS_freq_err_min>,<RMS_freq_err_max>`

Where

`<RMS_freq_err_avg>::<NRf>` is the average of RMS frequency error.

`<RMS_freq_err_min>::<NRf>` is the minimum of RMS frequency error.

`<RMS_freq_err_max>::<NRf>` is the maximum of RMS frequency error. Unit: Hz.

Examples

FETCH:PULSE:STATISTICS:MEASVIEW2:RMSFreqerror? might return 63.67E+3, 45.49E+3,81.28E+3 for the RMS frequency error in the pulse statistics.

FETCh:PULSe:STATistics[:MEASview<y>]:RMSPherror (Query Only)

Returns the RMS phase error in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT is set to TRENd.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Fetch commands

Syntax

FETCh:PULSe:STATistics[:MEASview<y>]:RMSPherror?

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<RMS_phase_err_avg>,<RMS_phase_err_min>,<RMS_phase_err_max>

Where

<RMS_phase_err_avg>::<NRf> is the average of RMS phase error.

<RMS_phase_err_min>::<NRf> is the minimum of RMS phase error.

<RMS_phase_err_max>::<NRf> is the maximum of RMS phase error. Unit: Degrees.

Examples

FETCH:PULSE:STATISTICS:MEASVIEW2:RMSPHERROR? might return 1.032,604.5E-3, 1.606 for the RMS phase error in the pulse statistics.

FETCh:PULSe:STATistics[:MEASview<y>]:RRATe (Query Only)

Returns the repetition rate in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT is set to TRENd.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Fetch commands

Syntax

FETCh:PULSe:STATistics[:MEASview<y>]:RRATe?

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<rep_rate_avg>,<rep_rate_min>,<rep_rate_max>

Where

<rep_rate_avg>::<NRf> is the average repetition rate.

<rep_rate_min>::<NRf> is the minimum repetition rate.

<rep_rate_max>::<NRf> is the maximum repetition rate. Unit: Hz.

Examples

FETCh:PULSe:STATistics[:MEASview<y>]:RRATe? might return 62.50E+3,62.49E+3, 62.52E+3 for the repetition rate in the pulse statistics.

FETCh:PULSe:STATistics[:MEASview<y>]:WIDTh (Query Only)

Returns the pulse width in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT is set to TRENd.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Fetch commands

Syntax

FETCh:PULSe:STATistics[:MEASview<y>]:WIDTh?

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<width_avg>,<width_min>,<width_max>

Where

<width_avg>::<NRf> is the average pulse width.

<width_min>::<NRf> is the minimum pulse width.

<width_max>::<NRf> is the maximum pulse width. Unit: Seconds.

Examples

FETCh:PULSe:STATistics[:MEASview<y>]:WIDTh? might return 4.636E-6,4.630E-6, 4.643E-6 for the pulse width in the pulse statistics.

FETCh:PULSe:STATistics[:MEASview<y>]:X? (Query Only)

Returns the horizontal values of the trace data of the Pulse Statistics measurement selected by the DISPlay:PULSe:SELect:RESult command.

NOTE: Select the plot type (Trend, FFT, Time Trend or Histogram) using the DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT command before executing this query.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Fetch commands

Syntax

FETCh:PULSe:STATistics[:MEASview<y>]:X?

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

#<num_digit><num_byte><data1><data2>...data<n>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the horizontal value of the STATistics:MEASview<y> graph at the nth data point.

The unit is trace number (Plot = Trend), Hz (Plot = FFT), second (Plot = Time Trend) or current measurement unit (Plot = Histogram).

4-byte little endian floating-point format specified in IEEE 488.2.

Examples

FETCh:PULSe:STATistics:MEASview2:X? might return #264xxxx. (64-byte data) for the STATistics:MEASview2 trace of the pulse width measurement when DISPlay:PULSe:SELect: RESult is set to WIDTh.

FETCh:PULSe:STATistics[:MEASview<y>]:XY? (Query Only)

Returns the horizontal and vertical values of the trace data of the pulse statistics measurement selected by the DISPlay:PULSe:SELect:RESult command.

NOTE: Select the plot type (Trend, FFT, Time Trend or Histogram) using the DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT command before executing this query.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Fetch commands

Syntax

FETCh:PULSe:STATistics[:MEASview<y>]:XY?

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

#<num_digit><num_byte><data1><data2>...data<n>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the horizontal value of the STATistics:MEASview<y> graph at the nth data point.

<data(n+1)> is the vertical value of the STATistics:MEASview<y> graph at the nth data point.

For horizontal values, the unit is trace number (Plot = Trend), Hz (Plot = FFT), second (Plot = Time Trend) or current measurement unit (Plot = Histogram).

For vertical values, the unit is current measurement unit (Plot = Trend or Time Trend), dB (Plot = FFT), count (Plot = Histogram)

4-byte little endian floating-point format specified in IEEE 488.2.

Examples

FETCh:PULSe:STATistics:MEASview1:XY? might return #264xxxx... (64-byte data) for the STATistics:MEASview1 trace of the pulse width measurement when DISPlay:PULSe:SELect: RESult is set to WIDTh.

FETCh:PULSe:STATistics[:MEASview<y>][:Y]? (Query Only)

Returns the trace data of the pulse statistics measurement selected by the DISPlay:PULSe:SELect:RESult command.

NOTE: Select the plot type (Trend, FFT, Time Trend or Histogram) using the DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT command before executing this query.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Fetch commands

Syntax

FETCh:PULSe:STATistics[:MEASview<y>][:Y]?

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

#<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the amplitude at the nth data point.

The unit is current measurement unit (Plot = Trend or Time Trend), dB (Plot = FFT), count or percent (Plot = Histogram)

4-byte little endian floating-point format specified in IEEE 488.2.

The unit of power is selected by the [SENSe]:POWer:UNITs command.

Examples

FETCH:PULSE:STATISTICS:MEASVIEW2:Y? might return #264xxxx... (64-byte data) for the STATistics 2 trace of the pulse width measurement when DISPlay:PULSe:SELect:RESult is set to WIDTh.

FETCh:PULSe:TRACe[:MEASview<y>]:X

Returns the time values of the pulse trace. Use the DISPlay:PULSe:SELect:NUMBer command to select the pulse, and the DISPlay:PULSe:SELect:RESult command to select the measurement result.

Conditions

Measurement views: Pulse trace with specified Measurement view number

Group

Fetch commands

Syntax

FETCh:PULSe:TRACe[:MEASview<y>]:X?

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

#<num_digit><num_byte><X(1)><X(2)>...<X(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<X(n)> is the time in seconds at the nth data point,

4-byte little endian floating-point format specified in IEEE 488.2.

Examples

FETCH:PULSE:TRACE:MEASVIEW1:X? might return #43204xxxx... (3204-byte data) for the time values of the trace in pulse trace measurement view 1.

FETCh:PULSe:TRACe[:MEASview<y>]:XY

Returns the horizontal (time) and vertical value pairs of the pulse trace. Use the DISPlay:PULSe:SELect:NUMBer command to select the pulse, and the DISPlay:PULSe:SELect:RESult command to select the measurement result.

Conditions

Measurement views: Pulse trace with specified Measurement view number

Group

Fetch commands

Syntax

`FETCh:PULSe:TRACe[:MEASview<y>]:XY?`

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`#<num_digit><num_byte><x(1)><y(1)><x(2)><y(2)>...<x(n)><y(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<x(n)><y(n)>` is the horizontal value (time in seconds) and vertical value pair at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

The vertical unit depends on measurement results: Hz for frequency error and deviation, degrees for phase error and deviation, otherwise dBm. The vertical unit can be changed by the `[SENSe]:POWer:UNITs` command.

Examples

`FETCH:PULSE:TRACE:MEASVIEW1:XY?` might return `#43204xxxx...` (3204-byte data) for the horizontal (time) and vertical value pairs of the pulse trace measurement view 1.

.

FETCh:PULSe:TRACe[:MEASview<y>]:Y

Returns the vertical values of the pulse trace. Use the `DISPlay:PULSe:SELect:NUMBer` command to select the pulse, and the `DISPlay:PULSe:SELect:RESult` command to select the measurement result.

Conditions

Measurement views: Pulse trace with specified Measurement view number

Group

Fetch commands

Syntax

`FETCh:PULSe:TRACe[:MEASview<y>]:Y?`

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`#<num_digit><num_byte><y(1)><y(2)>...<y(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

<y(n)> is the amplitude (dBm) at the nth data point,

4-byte little endian floating-point format specified in IEEE 488.2.

The unit depends on measurement results: Hz for frequency error and deviation, degrees for phase error and deviation, otherwise dBm. The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples

FETCH:PULSE:TRACE:MEASVIEW1:Y? might return #43204xxxx... (3204-byte data) for the vertical values of the pulse trace measurement view 1.

FETCh:RFIN:IQ? (Query Only)

Returns time-domain IQ data for a specific acquisition data record. You can set a range of IQ pairs optionally. The instrument needs to be in stopped mode. If not in stopped mode, it returns the execution error (-200).

Conditions

Measurement views: All

Group

Fetch commands

Syntax

FETCh:RFIN:IQ? <rec_ID>[,<begin_num>,<end_num>]

Arguments

<rec_ID> specifies the acquisition data record ID number.

<begin_num> specifies the beginning number of IQ pairs.

<end_num> specifies the end number of IQ pairs.

Use the FETCh:RFIN:RECord:IDS? query to get the beginning and end ID's of acquisition data records.

Returns

#<num_digit><num_byte><I(1)><Q(1)><I(2)><Q(2)>...<I(n)><Q(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<I(n)> and <Q(n)> are the time-domain IQ data pair.

4-byte little endian floating-point format specified in IEEE 488.2.

To find out the range of IQ pairs, use the FETCh:RFIN:IQ:HEADer? query. The returned value of <num_sample> is the number of IQ pairs.

NOTE: Do not fetch a large number of IQ pairs because of memory limitation.

Examples

FETCH:RFIN:IQ? 10,25,350 might return #43848xxxx... (3848-byte data) for the record #10, ranging from 25th to 350th IQ pair.

FETCh:RFIN:IQ:HEADer? (Query Only)

Returns the header information for a specific acquisition data record. The instrument needs to be in stopped mode. If not in stopped mode, it returns the execution error (-200).

Conditions

Measurement views: All

Group

Fetch commands

Syntax

`FETCh:RFIN:IQ:HEADer? <rec_ID>`

Arguments

`<rec_ID>` specifies the acquisition data record ID number.

Use the `FETCh:RFIN:RECord:IDS?` query to get the beginning and end ID's of acquisition data records.

Returns

`<span>,<sampling_freq>,<num_sample>,<center_freq>,<acq_BW>,`
`<ref_level>,<trig_pos>,<time_stamp>,<FastFrame_ID>`

Where

`<span>::<NR3>` is the span in Hz.

`<sampling_freq>::<NR3>` is the sampling frequency in Hz.

`<num_sample>::<NR3>` is the number of samples.

`<center_freq>::<NR3>` is the center frequency in Hz.

`<acq_BW>::<NR3>` is the acquisition bandwidth in Hz.

`<ref_level>::<NR3>` is the reference level in dBm.

`<trig_pos>::<NR3>` is the trigger position in seconds.

`<time_stamp>::<string>` is the time stamp.

`<FastFrame_ID>::<NR1>` is the Fast Frame ID: Zero or positive number. If the Fast Frame is disabled, the ID will be negative.

Examples

`FETCH:RFIN:IQ:HEADER? 10` might return `40.0E+6,50.0E+6,4.027E+3,`
`1.5E+9,40.0E+6,0.0,20.242E-6,"10/31/2007 1118:32 AM",-1`, indicating Span: 40 MHz, Sampling frequency: 50 MHz, Number of samples: 4027, Center frequency: 1.5 GHz, Acquisition bandwidth: 40 MHz, Reference level: 0 dBm, Trigger position: 20.242 µs, Time stamp: 10/31/2007 1118:32 AM, and Fast Frame ID: -1.

FETCh:RFIN:IQ:SCALe? (Query Only)

Returns the internal RF linear data scaling factor contained in the .tiq file header. The scaling factor can be used to convert digital IQ output (Option 05) values into real IQ values.

Conditions

Measurement views: All

Group

Fetch commands

Syntax

`FETCh:RFIN:IQ:SCALe?`

Related Commands

FETCh:RFIN:RECord:IDS? (on page 1359)

Arguments

None

Returns

`<NR3>` The RF linear data scaling factor.

Examples

`FETCH:RFIN:IQ:SCALE?` might return `19.553E-6` for the scaling factor.

FETCh:RFIN:IQ:VHEader? (Query Only)

Returns the verbose header information for a specific acquisition data record. The instrument needs to be in stopped mode. If not in stopped mode, it returns the execution error (-200).

Conditions

Measurement views: All

Group

Fetch commands

Syntax

`FETCh:RFIN:IQ:VHEader? <rec_ID>`

Arguments

`<rec_ID>` specifies the acquisition data record ID number.

Use the `FETCh:RFIN:RECord:IDS?` query to get the beginning and end ID's of acquisition data records.

Returns

```
<span>,<sampling_freq>,<num_sample>,<center_freq>,<acq_BW>,
<ref_level>,<trig_pos>,<time_stamp>,<FastFrame_ID>
```

Where

`<span>::<NR3>` is the span in Hz.

`<sampling_freq>::<NR3>` is the sampling frequency in Hz.

`<num_sample>::<NR3>` is the number of samples.

`<center_freq>::<NR3>` is the center frequency in Hz.

`<acq_BW>::<NR3>` is the acquisition bandwidth in Hz.

`<ref_level>::<NR3>` is the reference level in dBm.

`<trig_pos>::<NR3>` is the trigger position in seconds.

<time_stamp>::<string> is the time stamp.

<FastFrame_ID>::<NR1> is the Fast Frame ID: Zero or positive number. If the Fast Frame is disabled, the ID will be negative.

Examples

FETCH:RFIN:IQ:VHEADER? 10 might return 110.0000000000E+6,150.0000000000E+6,4.0270000000E+3, 2.7000000000E+9,110.0000000000E+6,0.0000,20.2420000000E-6, "7/9/2013 8:38:25.9305101 -07:00:00",-1, indicating Span: 110 MHz, Sampling frequency: 150 MHz, Number of samples: 4027, Center frequency: 2.7 GHz, Acquisition bandwidth: 110 MHz, Reference level: 0 dBm, Trigger position: 20.242 µs, Time stamp: 10/31/2007 1118:32 AM, Fast Frame ID: -1

FETCh:RFIN:RECord:IDS? (Query Only)

Returns the beginning and end ID numbers of acquisition data.

NOTE: The instrument needs to be in stopped mode. If not in stopped mode, it returns the execution error (-200).

Conditions

Measurement views: All

Group

Fetch commands

Syntax

FETCh:RFIN:RECord:IDS?

Arguments

None

Returns

<begin_ID>,<end_ID>

Where

<begin_ID>::<NR1> is the beginning ID of acquisition data.

<end_ID>::<NR1> is the end ID of acquisition data.

NOTE: "-1,-1" is returned when the span changes and the acquisition is armed, but the acquisition has not yet occurred.

Examples

FETCH:RFIN:RECORD:IDS? might return 1,147, indicating the beginning and end IDs of acquisition data are 1 and 147, respectively.

FETCh:SEM:MHITs:COUNt? (Query Only)

Returns the number of rows in the results table in the Spectral Emissions Mask measurement.

Conditions

Measurement views: Spectral Emissions Mask

Group

Fetch commands

Syntax

`FETCh:SEM:MHITs:COUNt?`

Arguments

None

Returns

`<NRf>` The number of rows in the table.

Examples

`FETCH:SEM:MHITS:COUNT?` might return `7`, indicating there are seven rows in the results table.

FETCh:SEM:MHITs<x>:AMPLitude:ABSolute? (Query Only)

Returns the absolute amplitude of the specified mask hit signal in the Spectral Emissions Mask measurement.

Conditions

Measurement views: Spectral Emissions Mask

Group

Fetch commands

Syntax

`FETCh:SEM:MHITs<x>:AMPLitude:ABSolute?`

Arguments

None

Returns

`<NRf>` Absolute amplitude of the specified signal in dBm.

`<x>` specifies the row in the results table, where the range is from 1 to 12.

Examples

`FETCh:SEM:MHITs2:AMPLitude:ABSolute?` might return `-73.17`, indicating that the absolute amplitude of signal in row 2 is –73.17 dBm.

FETCh:SEM:MHITs<x>:AMPLitude:RELative? (Query Only)

Returns the relative amplitude of the specified mask hit signal in the Spectral Emissions Mask measurement.

Conditions

Measurement views: Spectral Emissions Mask

Group

Fetch commands

Syntax

`FETCh:SEM:MHITs<x>:AMPLitude:RELative?`

Arguments

None

Returns

`<NRf>` Relative amplitude of the specified signal in dBm.

`<x>` specifies the row in the results table, where the range is from 1 to 12.

Examples

`FETCh:SEM:MHITs2:AMPLitude:RELative?` might return `-91.81`, indicating that the relative amplitude of mask hit signal in row 2 is –91.81 dBm.

FETCh:SEM:MHITs<x>:FREQuency? (Query Only)

Returns the frequency of the specified mask hit signal in the Spectral Emissions Mask measurement.

Conditions

Measurement views: Spectral Emissions Mask

Group

Fetch commands

Syntax

`FETCh:SEM:MHITs<x>:FREQuency?`

Arguments

None

Returns

`<NRf>` Relative amplitude of the specified signal in dBm.

`<x>` specifies the row in the results table, where the range is from 1 to 12.

Examples

`FETCh:SEM:MHITs4:FREQuency?` might return `2.235E+6`, indicating the frequency for mask hit signal in row 4 is 2.235 MHz.

FETCh:SEM:MHITs<x>:INTeg:ABSolute? (Query Only)

Returns the absolute integration amplitude of the specified mask hit signal in the Spectral Emissions Mask measurement.

Conditions

Measurement views: Spectral Emissions Mask

Group

Fetch commands

Syntax

FETCh:SEM:MHITs<x>:INTeg:ABSolute?

Arguments

None

Returns

<NRf> Absolute integration amplitude of the specified signal in dBm.

<x> specifies the row in the results table, where the range is from 1 to 12.

Examples

FETCh:SEM:MHITs4:INTeg:ABSolute? might return –75.14, indicating that the absolute integration amplitude for mask hit signal in row 4 is -75.14 dBM.

FETCh:SEM:MHITs<x>:INTeg:RELative? (Query Only)

Returns the relative integration amplitude of the specified mask hit signal in the Spectral Emissions Mask measurement.

Conditions

Measurement views: Spectral Emissions Mask

Group

Fetch commands

Syntax

FETCh:SEM:MHITs<x>:INTeg:RELative?

Arguments

None

Returns

<NRf> Relative integration amplitude of the specified signal in dBm.

<x> specifies the row in the results table, where the range is from 1 to 12.

Examples

FETCh:SEM:MHITs4:INTeg:RELative? might return –85.14, indicating that the relative integration amplitude for mask hit signal in row 4 is –85.14 dBM.

FETCh:SEM:MHITs<x>:MARGin:ABSolute? (Query Only)

Returns the absolute margin for the specified mask hit signal in the Spectral Emissions Mask measurement.

Conditions

Measurement views: Spectral Emissions Mask

Group

Fetch commands

Syntax

FETCh:SEM:MHITs<x>:MARGin:ABSolute?

Arguments

None

Returns

<NRf> Absolute margin of the specified signal in dB.

<x> specifies the row in the results table, where the range is from 1 to 12.

Examples

FETCh:SEM:MHITs4:MARGin:ABSolute? might return -62.33, indicating that the absolute margin for mask hit signal in row 4 is –62.33 dB.

FETCh:SEM:MHITs<x>:MARGin:RELative? (Query Only)

Returns the relative margin for the specified mask hit signal in the Spectral Emissions Mask measurement.

Conditions

Measurement views: Spectral Emissions Mask

Group

Fetch commands

Syntax

FETCh:SEM:MHITs<x>:MARGin:RELative?

Arguments

None

Returns

<NRf> Relative margin of the specified signal in dB.

<x> specifies the row in the results table, where the range is from 1 to 12.

Examples

FETCh:SEM:MHITs4:MARGin:RELative? might return -62.33, indicating that the relative margin for mask hit signal in row 4 is –62.33 .

FETCh:SEM:MHITs<x>:OFFSet? (Query Only)

Returns the offset for the specified mask hit signal in the Spectral Emissions Mask measurement.

Conditions

Measurement views: Spectral Emissions Mask

Group

Fetch commands

Syntax

FETCh:SEM:MHITs<x>:OFFSet?

Arguments

None

Returns

<NRf> specifies the mask hit offset.

<x> specifies the row in the results table, where the range is from 1 to 12.

Examples

FETCh:SEM:MHITs4:OFFSet? might return BL, indicating the offset for mask hit signal in row 4 is BL.

FETCh:SEM:PASS? (Query Only)

Returns the pass/fail limit test result in the Spectral Emissions Mask measurement.

Conditions

Measurement views: Spectral Emissions Mask

Group

Fetch commands

Syntax

FETCh:SEM:PASS?

Arguments

None

Returns

0 (fail) or 1 (pass)

Examples

FETCH:SEM:PASS? might return 1, indicating that the limit test was successful.

FETCh:SEM:REF:POWer? (Query Only)

Returns the reference power level in the Spectral Emissions Mask measurement.

Conditions

Measurement views: Spectral Emissions Mask

Group

Fetch commands

Syntax

FETCh:SEM:REF:POWer?

Arguments

None

Returns

<NRf> specifies the reference power.

Examples

`FETCH:SEM:REF:POWER?` might return `-16.92 dBm`, indicating that the reference power level is –16.92 dBm.

FETCh:SEM:SPECtrum:X? (Query Only)

Returns the frequencies of the spectrum trace in the Spectral Emissions Mask measurement.

Conditions

Measurement views: Spectral Emissions Mask

Group

Fetch commands

Syntax

`FETCh:SEM:SPECtrum:X?`

Arguments

None

Returns

`#<num_digit><num_byte><x(1)><x(2)>...<x(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<x(n)>` is the frequency (Hz) at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`FETCH:SEM:SPECTRUM:X?` might return `#516020xxxx...` (16020-byte data) for the frequencies of the spectrum trace.

FETCh:SEM:SPECtrum:XY? (Query Only)

Returns the frequency and amplitude pairs of the spectrum trace in the Spectral Emissions Mask measurement.

Conditions

Measurement views: Spectral Emissions Mask

Group

Fetch commands

Syntax

`FETCh:SEM:SPECtrum:XY?`

Arguments

None

Returns

`#<num_digit><num_byte><x(1)><y(1)><x(2)><y(2)>...<x(n)><y(n)>`

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<x(n)><y(n)> is the frequency (Hz) and amplitude (dBm) pair at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

FETCH:SEM:SPECTRUM:XY? might return #516020xxxx... (16020-byte data) for the frequency and amplitude pairs of the spectrum trace.

FETCh:SEM:SPECtrum[:Y]? (Query Only)

Returns the amplitude of the spectrum trace in the Spectral Emissions Mask measurement.

Conditions

Measurement views: Spectral Emissions Mask

Group

Fetch commands

Syntax

FETCh:SEM:SPECtrum[:Y]?

Arguments

None

Returns

#<num_digit><num_byte><x(1)><x(2)>...<x(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<x(n)> is the frequency (Hz) at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

FETCH:SEM:SPECTRUM[:Y]? might return #516020xxxx... (16020-byte data) for the frequencies of the spectrum trace.

FETCh:SGRam[:MEASview<y>](Query Only)

Returns trace data of a line in the spectrogram. The line is selected using the TRACe:SGRam[:MEASview<y>]:SELect:LINE command.

Conditions

Measurement views: Spectrogram

Group

Fetch commands

Syntax

FETCh:SGRam[:MEASview<y>]?

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the amplitude in dBm at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2. The unit can be changed by the `[SENSe]:POWer:UNITs` command.

Examples

`FETCh:SGRam:MEASview1?` might return `#43204xxxx...` (3204-byte data) for the line in the spectrogram.

FETCh:SIGNalvu:ACQuisition:COUPling? (Query Only)

Returns the coupling value of the acquisition.

Conditions

Should be connected to MSO Query

Group

Fetch commands

Syntax

`FETCh:SIGNalvu:ACQuisition:COUPling?`

Arguments

None

Returns

`<max>::<NRf>` is the maximum phase in degrees.

Examples

`FETCh:SIGNalvu:ACQuisition:COUPling?` might return 153.8, indicating the maximum phase is 153.8 °.

FETCh:SIGNalvu:ACQuisition:MAX:RT:BW? (Query Only)

Queries the maximum real time bandwidth value of the instrument.

Conditions

Should be connected to MSO Query

Group

Fetch commands

Syntax
FETCh:SIGNalvu:ACQuisition:MAX:RT:BW

Related Commands
FETCh:SIGNalvu:ACQuisition:MAX:RT:BW?

Arguments
None

Returns
<max>::<NRf> is the maximum phase in degrees.

Examples
FETCh:SIGNalvu:ACQuisition:MAX:RT:BW? might return 153.8, indicating the maximum phase is 153.8 °.

FETCH:SIGNalvu:ACQuisition[:SOURce<x>]:DOWNconverter:INJEction:HIGHSide:MAXFreq? (Query only)
Returns the maximum range of input center frequency in case of high injection side. The parameter <x>=1 to 8 represents the range of the source.

Conditions
Should be connected to MSO/LPD/RSA/SX to query.

Source keyword is optional, if not present Source1 will be default source.

Group
Fetch commands

Syntax
FETCH:SIGNalvu:ACQuisition[:SOURce<x>]:DOWNconverter:INJEction:HIGHSide:MAXFreq?

Related Commands
:FETCH:SIGNalvu:ACQuisition[:SOURce<x>]:DOWNconverter:INJEction:HIGHSide:MINFreq? (on page 1368)

Arguments
None

Returns
<max>::<NRf> returns the maximum value of the centre frequency range in case of high injection side.

Examples
:FETCH:SIGNalvu:ACQuisition:SOURce1:DOWNconverter:INJEction:HIGHSide:MAXFreq? might return 9.5000000000E+9, indicating the maximum range of the input center frequency in case of high injection side.

FETCH:SIGNalvu:ACQuisition[:SOURce<x>]:DOWNconverter:INJEction:HIGHSide:MINFreq? (Query only)
Returns the minimum range of the input center frequency in case of high injection side. The parameter <x>=1 to 8 represents the range of the source.

Conditions

Should be connected to MSO/LPD/RSA/SX to query.

Source keyword is optional, if not present, Source1 is default source.

Group

Fetch commands

Syntax

`FETCH:SIGNalvu:ACQuisition[:SOURce<x>]:DOWNconverter:INJEction:HIGHSide:MINFreq?`

Arguments

None

Returns

`<max>::<NRf>` returns the minimum value of the center frequency range in case of high injection side.

Examples

`:FETCH:SIGNalvu:ACQuisition:SOURce1:DOWNconverter:INJEction:HIGHSide:MINFreq?` might return `2.0000000000E+9`, indicating the minimum range of the input center frequency in case of high injection side.

FETCH:SIGNalvu:ACQuisition[:SOURce<x>]:DOWNconverter:INJEction:LOWSide:MAXFreq? (Query only)

Returns the maximum range of the input center frequency in case of low injection side. The parameter <x>=1 to 8 represents the range of the source.

Conditions

Should be connected to MSO/LPD/RSA/SX to query.

Source keyword is optional, if not present, Source1 will be default source.

Group

Fetch commands

Syntax

`FETCH:SIGNalvu:ACQuisition[:SOURce<x>]:DOWNconverter:INJEction:LOWSide:MAXFreq?`

Related Commands

FETCH:SIGNalvu:ACQuisition[:SOURce<x>]:DOWNconverter:INJEction:LOWSide:MINFreq? (on page 1370)

Arguments

None

Returns

`<max>::<NRf>` returns the maximum value of centre frequency range in case of low injection side.

Examples

`:FETCH:SIGNalvu:ACQuisition:SOURce1:DOWNconverter:INJEction:LOWSide:MAXFreq?` Might return `9.5000000000E+9`, indicating the maximum range of the input center frequency in case of low injection side.

FETCH:SIGNalvu:ACQuisition[:SOURce<x>]:DOWNconverter:INJEction:LOWSide:MINFreq? (Query only)

Returns the minimum range of input center frequency in case of low injection side. The parameter <x>=1 to 8 represents the range of the source.

Conditions

Should be connected to MSO/LPD/RSA/SX to query.

Source keyword is optional, if not present, Source1 will be default source

Group

Fetch commands

Syntax
`FETCH:SIGNalvu:ACQuisition[:SOURce<x>]:DOWNconverter:INJEction:LOWSide:MINFreq?`

Arguments

None

Returns

`<max>::<NRf>` returns the minimum value of the centre frequency range in case of low injection side.

Examples

`:FETCH:SIGNalvu:ACQuisition:SOURce1:DOWNconverter:INJEction:LOWSide:MINFreq?` might return `2.0000000000E+9`, indicating the minimum range of the input center frequency in case of low injection side.

FETCh:SIGNalvu:ACQuisition:TERMination? (Query Only)

Queries the termination value of the acquisition.

Conditions

Should be connected to MSO Query

Group

Fetch commands

Syntax

`FETCh:SIGNalvu:ACQuisition:TERMination?`

Arguments

None

Returns

`<max>::<NRf>` is the maximum phase in degrees.

Examples

`FETCh:SIGNalvu:ACQuisition:TERMination?` might return 153.8, indicating the maximum phase is 153.8 °.

FETCh:SPECtrum[:MEASview<y>]:TRACe:BITMap? (Query Only)

Returns trace waveform data of the bitmap trace in the Spectrum measurement.

Conditions

Measurement views: Spectrum

Group

Fetch commands

Syntax

`FETCh:SPECtrum[:MEASview<y>]:TRACe:BITMap?`

Arguments

None

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Returns

A binary block of 32-bit floating point values (IEEE standard single precision floating point format). Each value ranges from 0 to 1.0, where 1.0 represents 100%.

Bitmaps have 201 rows. Data[0] is the first column in the first row, and the following data continues along that row before moving to the next.

Bitmap values are returned in the order of top left pixel first, then the rest of the top row, after that row by row.
In other words, the pixels in the bitmap are returned from left to right and top to bottom; one row at a time.

Examples

`FETCH:SPECTRUM:TRACE:BITMAP?` might return #6644004xxxx...

FETCh:SPECtrum[:MEASview<y>]:TRACe<x>? (Query Only)

Returns the trace data in the Spectrum measurement.

The parameter <x> = 1 to 5.

NOTE: TRACe5 (spectrogram) is valid when the spectrum and spectrogram measurements are running.

Conditions

Measurement views: Spectrum

Group

Fetch commands

Syntax

`FETCh:SPECtrum[:MEASview<y>]:TRACe<x>?`

Related Commands

TRACe<x>:SPECtrum (on page 2684)

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Returns

#<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the amplitude in dBm at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2. The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples

FETCH:SPECTRUM:MEASview1:TRACE1? might return #43204xxxx... (3204-byte data) for Trace 1 in the Spectrum measurement.

FETCh:SPURious:CARRier:POWer? (Query Only)

Returns the carrier power in the Spurious measurement.

Conditions

Measurement views: Spurious

Group

Fetch commands

Syntax

FETCh:SPURious:CARRier:POWer?

Arguments

None

Returns

<NRf> Carrier power in dBm.

The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples

FETCH:SPURIOUS:CARRIER:POWER? might return 4.227, indicating that the carrier power is 4.227 dBm.

FETCh:SPURious:COUNt? (Query Only)

Returns the number of spurious signals in the Spurious measurement.

Conditions

Measurement views: Spurious

Group

Fetch commands

Syntax
`FETCh:SPURious:COUNt?`

Arguments
None

Returns
`<NRf>` The number of spurious signals.

Examples
`FETCH:SPURIOUS:COUNT?` might return `4`, indicating that the spurious count is 4.

FETCh:SPURious:PASS? (Query Only)
Returns the pass/fail limit test result in the Spurious measurement.

Conditions
Measurement views: Spurious

Group
Fetch commands

Syntax
`FETCh:SPURious:PASS?`

Arguments
None

Returns
0 (fail) or 1 (pass).

Examples
`FETCH:SPURIOUS:PASS?` might return `1`, indicating that the limit test was successful.

FETCh:SPURious:SPECtrum:X? (Query Only)
Returns the frequencies of the spectrum trace in the Spurious measurement.

Conditions
Measurement views: Spurious

Group
Fetch commands

Syntax
`FETCh:SPURious:SPECtrum:X?`

Arguments
None

Returns

#<num_digit><num_byte><x(1)><x(2)>...<x(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<x(n)> is the frequency (Hz) at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

FETCH:SPURIOUS:SPECTRUM:X? might return #516020xxxx... (16020-byte data) for the frequencies of the spectrum trace in the Spurious measurement.

FETCh:SPURious:SPECtrum:XY? (Query Only)

Returns the frequency and amplitude pairs of the spectrum trace in the Spurious measurement.

Conditions

Measurement views: Spurious

Group

Fetch commands

Syntax

FETCh:SPURious:SPECtrum:XY?

Arguments

None

Returns

#<num_digit><num_byte><x(1)><y(1)><x(2)><y(2)>...<x(n)><y(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<x(n)><y(n)> is the frequency (Hz) and amplitude (dBm) pair at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

The amplitude unit can be changed by the [SENSe]:POWer:UNITs command.

Examples

FETCH:SPURIOUS:SPECTRUM:XY? might return #516020xxxx... (16020-byte data) for the frequency and amplitude pairs of the spectrum trace in the Spurious measurement.

FETCh:SPURious:SPECtrum[:Y]? (Query Only)

Returns the amplitudes of the spectrum trace in the Spurious measurement.

Conditions

Measurement views: Spurious

Group

Fetch commands

Syntax

`FETCh:SPURious:SPECtrum[:Y]?`

Arguments

None

Returns

`#<num_digit><num_byte><y(1)><y(2)>...<y(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<y(n)>` is the amplitude (dBm) at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2. The unit can be changed by the `[SENSe]:POWer:UNITs` command.

Examples

`FETCH:SPURIOUS:SPECTRUM:Y?` might return `#516020xxxx...` (16020-byte data) for the amplitudes of the spectrum trace in the Spurious measurement.

FETCh:SPURious:SPUR<x>:AMPLitude:ABSolute? (Query Only)

Returns the absolute amplitude of the specified spurious signal in the Spurious measurement.

Conditions

Measurement views: Spurious

Group

Fetch commands

Syntax

`FETCh:SPURious:SPUR<x>:AMPLitude:ABSolute?`

Arguments

None

Returns

`<NRf>` Absolute amplitude of the specified spurious signal in dBm. The unit can be changed by the `[SENSe]:POWer:UNITs` command.

Examples

`FETCH:SPURIOUS:SPUR1:AMPLITUDE:ABSOLUTE?` might return `-19.782`, indicating that the absolute amplitude of Spurious #1 is −19.782 dBm.

FETCh:SPURious:SPUR<x>:AMPLitude:RELative? (Query Only)

Returns the relative amplitude of the specified spurious signal in the Spurious measurement.

Conditions

Measurement views: Spurious

Group

Fetch commands

Syntax

`FETCh:SPURious:SPUR<x>:AMPLitude:RELative?`

Arguments

None

Returns

`<NRf>` Relative amplitude of the specified spurious signal in dB. Use the `[SENSe]:SPURious:REFerence` command to set the power reference.

Examples

`FETCH:SPURIOUS:SPUR1:AMPLITUDE:RELATIVE?` might return `-9.782`, indicating that the relative amplitude of Spurious #1 is -9.782 ndB.

FETCh:SPURious:SPUR<x>:FREQuency:ABSolute? (Query Only)

Returns the absolute frequency of the specified spurious signal in the Spurious measurement.

Conditions

Measurement views: Spurious

Group

Fetch commands

Syntax

`FETCh:SPURious:SPUR<x>:FREQuency:ABSolute?`

Arguments

None

Returns

`<NRf>` Absolute frequency of the spurious signal in Hz.

Examples

`FETCH:SPURIOUS:SPUR1:FREQUENCY:ABSOLUTE?` might return `2.235E+9`, indicating that the absolute frequency of Spurious #1 is 2.235 GHz.

FETCh:SPURious:SPUR<x>:FREQuency:RELative? (Query Only)

Returns the relative frequency of the specified spurious signal to the carrier in the Spurious measurement. This command is valid when `[SENSe]:SPURious:REFerence` is set to CARRier.

Conditions

Measurement views: Spurious

Group

Fetch commands

Syntax

```
FETCh:SPURious:SPUR<x>:FREQuency:RELative?
```

Arguments

None

Returns

<NRf> Relative frequency of the spurious signal to the carrier in Hz.

Examples

FETCH:SPURIOUS:SPUR1:FREQUENCY:RELATIVE might return 3.634E+6, indicating that the relative frequency of Spurious #1 is 3.634 MHz.

FETCh:SPURious:SPUR<x>:LIMit:ABSolute? (Query Only)

Returns the absolute amplitude of the limit for the specified spurious signal in the Spurious measurement.

Conditions

Measurement views: Spurious

Group

Fetch commands

Syntax

```
FETCh:SPURious:SPUR<x>:LIMit:ABSolute?
```

Arguments

None

Returns

<NRf> Absolute amplitude of the limit for the specified spurious signal in dBm.

The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples

FETCH:SPURIOUS:SPUR1:LIMIT:ABSOLUTE? might return -50.0, indicating that the absolute amplitude of the limit for Spurious #1 is -50 dBm.

FETCh:SPURious:SPUR<x>:LIMit:RELative? (Query Only)

Returns the relative amplitude of the limit for the specified spurious signal in the Spurious measurement.

Conditions

Measurement views: Spurious

Group

Fetch commands

Syntax
`FETCh:SPURious:SPUR<x>:LIMit:RELative?`

Arguments
None

Returns
`<NRf>` Relative amplitude of the limit for the specified spurious signal in dB. Use the `[SENSe]:SPURious:REFerence` command to set the power reference.

Examples
`FETCH:SPURIOUS:SPUR1:LIMIT:RELATIVE?` might return `-10.0`, indicating that the relative amplitude of the limit for Spurious #1 is -10 dB.

FETCh:SPURious:SPUR<x>:LIMit:VIOLation? (Query Only)
Returns whether the specified spurious signal exceeds the limit or not.

Conditions
Measurement views: Spurious

Group
Fetch commands

Syntax
`FETCh:SPURious:SPUR<x>:LIMit:VIOLation?`

Arguments
None

Returns
`0` (under the limit) or `1` (over the limit).

Examples
`FETCH:SPURIOUS:SPUR1:LIMIT:VIOLATION?` might return 1, indicating that Spurious #1 exceeds the limit.

FETCh:SPURious:SPUR<x>:RANGe? (Query Only)
Returns the frequency range in which the specified spurious signal occurred.

Conditions
Measurement views: Spurious

Group
Fetch commands

Syntax
`FETCh:SPURious:SPUR<x>:RANGe?`

Arguments

None

Returns

<string> "A" to "T" representing Range A to T, respectively.

Examples

FETCH:SPURIOUS:SPUR1:RANGE might return "E", indicating that Spurious #1 is in Range E.

FETCh:SQUality[:MEASview<y>]:FREQuency:DEViation? (Query only)

Returns the frequency deviation in the signal quality measurement.

This command is valid when [SENSe]:DDEMod:MODulation:TYPE is set to C4FM, FSK2, FSK4, FSK8, or FSK16.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Fetch commands

Syntax

FETCh:SQUality[:MEASview<y>]:FREQuency:DEViation?

Arguments

None

Returns

<NRf> Frequency deviation in Hz.

Examples

FETCh:SQUality:MEASview1:FREQuency:DEViation? might return 12.68E+3, indicating the frequency deviation is 12.68 kHz for measurement view1.

FETCh:SQUality[:MEASview<y>]:FREQuency:DEViation:TABLe? (Query only)

Returns the number of columns and the values in the frequency deviation table for a signal quality measurement.

This command is valid when [SENSe]:DDEMod:MODulation:TYPE is set to C4FM, FSK2, FSK4, FSK8, or FSK16.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Fetch commands

Syntax

`FETCh:SQUality[:MEASview<y>]:FREQuency:DEViation:TABLe?`

Arguments

None

Returns

`<Dev_Num>,{<Freq_dev_Max>,<Freq_dev_Min>,<Freq_dev_Avg>}`

Where

`<Dev_Num>::<NR1>` is the number of columns in the frequency deviation table.

= 2 (2FSK), 4 (4FSK, C4FM), 8 (8FSK), or 16 (16FSK)

`<Freq_dev_Max>::<NRf>` is the maximum frequency deviation in Hz.

`<Freq_dev_Min>::<NRf>` is the minimum frequency deviation in Hz.

`<Freq_dev_Avg>::<NRf>` is the average frequency deviation in Hz.

The dataset `<Freq_dev_Max>,<Freq_dev_Min>,<Freq_dev_Avg>` is returned for each symbol in ascending order of its level (for example, in order of symbol -3, -1, +1, and +3 for 4FSK).

Examples

`FETCh:SQUality:MEASview1FREQuency:DEVIation:TABLe?` might return 2,1.257E+3,1.039E+3,1.162E+3,1.586E+3,1.298E+3,1.425E+3 for the frequency signal, populating the results table as follows.

Deviations	-1	+1
Maximum	1.257 kHz	1.586 kHz
Minimum	1.039 kHz	1.298 kHz
Average	1.162 kHz	1.425 kHz

FETCh:SQUality[:MEASview<y>]:FREQuency:ERRor? (Query only)

Returns the frequency error in the signal quality measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Fetch commands

Syntax

`FETCh:SQUality[:MEASview<y>]:FREQuency:ERRor?`

Arguments

None

Returns

`<NRf>` Frequency error in Hz.

Examples

`FETCh:SQUality:MEASview1:FREQuency:ERRor?` might return 612.043E+3, indicating that the frequency error is 612.043 kHz for measurement view 1.

FETCh:SQUality[:MEASview<y>]:GAIN:IMBalance? (Query only)

Returns the gain imbalance in the signal quality measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Fetch commands

Syntax

`FETCh:SQUality[:MEASview<y>]:GAIN:IMBalance?`

Arguments

None

Returns

`<NRf>` Gain imbalance in dB.

Examples

`FETCh:SQUality[:MEASview1:GAIN:IMBalance?` might return -57.746E-3, indicating that the gain imbalance is -0.057746 dB for measurement view 1.

FETCh:SQUality[:MEASview<y>]:ORIGin:OFFSet? (Query only)

Returns the origin offset in the signal quality measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Fetch commands

Syntax

`FETCh:SQUality[:MEASview<y>]:ORIGin:OFFSet?`

Arguments

None

Returns

`<NRf>` Origin offset in dB.

Examples

`FETCh:SQUality:MEASview1:ORIGin:OFFSet?` might return -44.968, indicating that the origin offset is -44.968 dB for measurement view1.

FETCh:SQUality[:MEASview<y>]:PEAK:EVM? (Query only)

Returns the peak EVM (%) in the signal quality measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Fetch commands

Syntax

`FETCh:SQUality[:MEASview<y>]:PEAK:EVM?`

Arguments

None

Returns

`<NRf>` Peak EVM in percent (%).

Examples

`FETCh:SQUality:MEASview1:PEAK:EVM?` might return 4.276, indicating that the peak EVM is 4.276% for measurement view1.

FETCh:SQUality[:MEASview<y>]:PEAK:EVM:DB? (Query only)

Returns the peak EVM (dB) in the signal quality measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Fetch commands

Syntax

`FETCh:SQUality[:MEASview<y>]:PEAK:EVM:DB?`

Arguments

None

Returns

`<NRf>` Peak EVM in dB.

Examples

`FETCh:SQUality:MEASview1:PEAK:EVM:DB?` might return -27.358, indicating that the peak EVM is -27.358 dB for measurement view 1.

FETCh:SQUality[:MEASview<y>]:PEAK:EVM:DB:OFFSet? (Query only)

Returns the peak offset EVM (dB) in the signal quality measurement.

This command is valid when `[SENSe]:DDEMod:MODulation:TYPE` is set to OQPSK or SOQPSK.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Fetch commands

Syntax

`FETCh:SQUality[:MEASview<y>]:PEAK:EVM:DB:OFFSet?`

Arguments

None

Returns

`<NRf>` Peak offset EVM in dB.

Examples

`FETCh:SQUality:MEASview1:PEAK:EVM:DB:OFFSet?` might return –37.624, indicating the peak offset EVM is -37.624 dB for measurement view 1.

FETCh:SQUality[:MEASview<y>]:PEAK:EVM:LOCation? (Query only)

Returns the time at which the EVM is peak.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Fetch commands

Syntax

`FETCh:SQUality[:MEASview<y>]:PEAK:EVM:LOCation?`

Arguments

None

Returns

`<NRf>` The time in symbol number at which the EVM is peak.

The unit can be changed by the `[SENSe]:DDEMod:TIMe:UNITs` command.

Examples

`FETCh:SQUality:MEASview1:PEAK:EVM:LOCation?` might return 68.000, indicating that the EVM is peak at symbol #68.000 for measurement view1.

FETCh:SQUality[:MEASview<y>]:PEAK:EVM:LOCation:OFFSet? (Query only)

Returns the time at which the offset EVM is peak.

This command is valid when `[SENSe]:DDEMod:MODulation:TYPE` is set to OQPSK or SOQPSK.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Fetch commands

Syntax

`FETCh:SQUality[:MEASview<y>]:PEAK:EVM:LOCation:OFFSet?`

Arguments

None

Returns

`<NRf>` The time in symbol number at which the offset EVM is peak.

The unit can be changed by the `[SENSe]:DDEMod:TIMe:UNITs` command.

Examples

`FETCh:SQUality:MEASview1:PEAK:EVM:LOCation:OFFSet?` might return 123.00, indicating that the offset EVM is peak at symbol #123 for measurement view1.

FETCh:SQUality[:MEASview<y>]:PEAK:EVM:OFFSet? (Query only)

Returns the peak offset EVM (%) in the signal quality measurement.

This command is valid when `[SENSe]:DDEMod:MODulation:TYPE` is set to OQPSK or SOQPSK.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Fetch commands

Syntax

`FETCh:SQUality[:MEASview<y>]:PEAK:EVM:OFFSet?`

Arguments

None

Returns

`<NRf>` Peak offset EVM in percent (%).

Examples

`FETCh:SQUality:MEASview1:PEAK:EVM:OFFSet?` might return 1.298, indicating the peak offset EVM is 1.298% for measurement view1.

FETCh:SQUality[:MEASview<y>]:PEAK:FERRor? (Query only)

Returns the peak FSK error in the signal quality measurement.

This command is valid when `[SENSe]:DDEMod:MODulation:TYPE` is set to FSK2, FSK4, FSK8, or FSK16.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Fetch commands

Syntax

`FETCh:SQUality[:MEASview<y>]:PEAK:FERRor?`

Arguments

None

Returns

`<NRf>` Peak FSK error in percent (%).

Examples

`FETCh:SQUality:MEASview1:PEAK:FERRor?` might return 9.136, indicating the peak FSK error is 9.136% for measurement view1.

FETCh:SQUality[:MEASview<y>]:PEAK:MERRor:DB? (Query only)

Returns the peak magnitude error (dB) in the signal quality measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Fetch commands

Syntax

`FETCh:SQUality[:MEASview<y>]:PEAK:MERRor:DB?`

Arguments

None

Returns

`<NRf>` Peak magnitude error in dB.

Examples

`FETCh:SQUality:MEASview1:PEAK:EVM:DB?` might return -27.358, indicating that the peak EVM is -27.358 dB for measurement view1.

FETCh:SQUality[:MEASview<y>]:PEAK:MERRor:LOCation? (Query only)

Returns the time at which the magnitude error is peak.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Fetch commands

Syntax

`FETCh:SQUality[:MEASview<y>]:PEAK:MERRor:LOCation?`

Arguments

None

Returns

`<NRf>` The time in symbol number at which the magnitude error is peak.

The unit can be changed by the `[SENSe]:DDEMod:TIMe:UNITs` command.

Examples

`FETCh:SQUality:MEASview1:PEAK:MERRor:LOCation?` might return 68.000, indicating that the magnitude error is peak at symbol #68 for measurement view1.

FETCh:SQUality[:MEASview<y>]:PEAK:MERRor? (Query only)

Returns the peak magnitude error (%) in the signal quality measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Fetch commands

Syntax

`FETCh:SQUality[:MEASview<y>]:PEAK:MERRor?`

Arguments

None

Returns

<NRf> Peak magnitude error in percent (%).

Examples

FETCh:SQUality:MEASview1:PEAK:MERRor? might return 3.595, indicating that the peak magnitude error is 3.595% for measurement view1.

FETCh:SQUality[:MEASview<y>]:PEAK:PERRor:LOCation? (Query only)

Returns the time at which the phase error is peak.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Fetch commands

Syntax

FETCh:SQUality[:MEASview<y>]:PEAK:PERRor:LOCation?

Arguments

None

Returns

<NRf> The time in symbol number at which the phase error is peak.

The unit can be changed by the [SENSe]:DDEMod:TIMe:UNITs command.

Examples

FETCh:SQUality:MEASview1:PEAK:PERRor:LOCation? might return 68.000, indicating that the phase error is peak at symbol #68 for measurement view1.

FETCh:SQUality[:MEASview<y>]:PEAK:PERRor? (Query only)

Returns the peak phase error in the signal quality measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Fetch commands

Syntax

`FETCh:SQUality[:MEASview<y>]:PEAK:PERRor?`

Arguments

None

Returns

`<NRf>` Peak phase error in degrees.

Examples

`FETCh:SQUality:MEASview1:PEAK:PERRor?` might return 1.907, indicating that the peak phase error is 1.907 ° for measurement view1.

FETCh:SQUality[:MEASview<y>]:QUADrature:ERRor? (Query only)

Returns the quadrature error in the signal quality measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Fetch commands

Syntax

`FETCh:SQUality[:MEASview<y>]:QUADrature:ERRor?`

Arguments

None

Returns

`<NRf>` Quadrature error in degrees.

Examples

`FETCh:SQUality:MEASview1:QUADrature:ERRor?` might return -14.264E-3, indicating that the quadrature error is -0.014264° for measurement view1.

FETCh:SQUality[:MEASview<y>]:RHO? (Query only)

Returns the ρ (waveform quality) value in the signal quality measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Fetch commands

Syntax

`FETCh:SQUality[:MEASview<y>]:RHO?`

Arguments

None

Returns

`<NRf>` ρ value.

Examples

`FETCh:SQUality:MEASview1:RHO?` might return 998.703E-3, indicating that ρ is 0.998703 for measurement view1.

FETCh:SQUality[:MEASview<y>]:RMS:EVM? (Query only)

Returns the RMS EVM (%) in the signal quality measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Fetch commands

Syntax

`FETCh:SQUality[:MEASview<y>]:RMS:EVM?`

Arguments

None

Returns

`<NRf>` RMS EVM in percent (%).

Examples

`FETCh:SQUality:MEASview1:RMS:EVM?` might return 2.417, indicating that the RMS EVM is 2.417% for measurement view 1.

FETCh:SQUality[:MEASview<y>]:RMS:EVM:DB? (Query only)

Returns the RMS EVM (dB) in the signal quality measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Fetch commands

Syntax

`FETCh:SQUality[:MEASview<y>]:RMS:EVM:DB?`

Arguments

None

Returns

`<NRf>` RMS EVM in dB.

Examples

`FETCh:SQUality:MEASview1:RMS:EVM:DB?` might return -32.356, indicating that the RMS EVM is -32.356 dB for measurement view 1.

FETCh:SQUality[:MEASview<y>]:RMS:EVM:DB:OFFSet? (Query only)

Returns the RMS offset EVM (dB) in the signal quality measurement.

This command is valid when `[SENSe]:DDEMod:MODulation:TYPE` is set to OQPSK or SOQPSK.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Fetch commands

Syntax

`FETCh:SQUality[:MEASview<y>]:RMS:EVM:DB:OFFSet?`

Arguments

None

Returns

`<NRf>` RMS offset EVM in dB.

Examples

`FETCh:SQUality:MEASview1:RMS:EVM:DB:OFFSet?` might return –41.276, indicating the RMS offset EVM is -41.276 dB for measurement view 1.

FETCh:SQUality[:MEASview<y>]:RMS:EVM:OFFSet? (Query only)

Returns the RMS offset EVM (%) in the signal quality measurement.

This command is valid when `[SENSe]:DDEMod:MODulation:TYPE` is set to OQPSK or SOQPSK.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Fetch commands

Syntax

`FETCh:SQUality[:MEASview<y>]:RMS:EVM:OFFSet?`

Arguments

None

Returns

`<NRf>` RMS offset EVM in percent (%).

Examples

`FETCh:SQUality:MEASview1:RMS:EVM:OFFSet?` might return 0.783, indicating the RMS offset EVM is 0.783% for measurement view 1.

FETCh:SQUality[:MEASview<y>]:RMS:FERRor? (Query only)

Returns the RMS FSK error percent in the signal quality measurement.

This command is valid when `[SENSe]:DDEMod:MODulation:TYPE` is set to FSK2, FSK4, FSK8, or FSK16.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Fetch commands

Syntax

`FETCh:SQUality[:MEASview<y>]:RMS:FERRor?`

Arguments

None

Returns

<NRf> RMS FSK error in percent (%).

Examples

FETCh:SQUality:MEASview1:RMS:FERRor? might return 1.815, indicating the RMS FSK error is 1.815% for measurement view 1.

FETCh:SQUality[:MEASview<y>]:RMS:MER:DB? (Query only)

Returns the RMS MER (Modulation Error Ratio) in dB in the signal quality measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Fetch commands

Syntax

FETCh:SQUality[:MEASview<y>]:RMS:MER:DB?

Arguments

None

Returns

<NRf> RMS MER in dB.

Examples

FETCh:SQUality:MEASview1:RMS:MER:DB? might return 27.394, indicating that the RMS MER is 27.394 dB for measurement view1.

FETCh:SQUality[:MEASview<y>]:RMS:MERRor? (Query only)

Returns the RMS magnitude error (%) in the signal quality measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Fetch commands

Syntax

`FETCh:SQUality[:MEASview<y>]:RMS:MERRor?`

Arguments

None

Returns

`<NRf>` RMS magnitude error in percent (%).

Examples

`FETCh:SQUality:MEASview1:RMS:MERRor?` might return 1.837, indicating that the RMS magnitude error is 1.837% for measurement view 1.

FETCh:SQUality[:MEASview<y>]:RMS:MERRor:DB? (Query only)

Returns the RMS magnitude error (dB) in the signal quality measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Fetch commands

Syntax

`FETCh:SQUality[:MEASview<y>]:RMS:MERRor:DB?`

Arguments

None

Returns

`<NRf>` RMS MERRor in dB.

Examples

`FETCh:SQUality:MEASview1:RMS:MERRor:DB?` might return -34.706, indicating that the magnitude error is -34.706 dB for measurement view 1.

FETCh:SQUality[:MEASview<y>]:RMS:PERRor? (Query only)

Returns the RMS phase error in the signal quality measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Fetch commands

Syntax

`FETCh:SQUality[:MEASview<y>]:RMS:PERRor?`

Arguments

None

Returns

`<NRf>` RMS phase error in degrees.

Examples

`FETCh:SQUality:MEASview1:RMS:PERRor?` might return 893.472E-3, indicating that the RMS phase error is 0.893472 °.

FETCh:SQUality[:MEASview<y>]:SYMBol:LENGth? (Query only)

Returns the number of analyzed symbols.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Fetch commands

Syntax

`FETCh:SQUality[:MEASview<y>]:SYMBol:LENGth?`

Returns

`<NR1>` Indicates the length of the synch word in symbols.

Examples

`FETCh:SQUality:MEASview1:SYMBol:LENGth?` might return 3, indicating the length is three symbols for measurement view 1.

FETCh:SQUality[:MEASview<y>]:SYMBol:RATE? (Query only)

Returns the value of the calculated symbol rate in Hz.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

It is valid when the modulation type is 2|4|8|16FSK and `[:SENSe]:DDEMod:SYMBol:RATE:SEARch` is ON.

Group

Fetch commands

Syntax

`FETCh:SQUality[:MEASview<y>]:SYMBol:RATE?`

Related Commands

[SENSe]:DDEMod[:SOURce<x>]:SYMBol:RATE:SEARch (on page 2047)

Returns

`<NRf>` is the calculated symbol rate in Hz.

Examples

`FETCh:SQUality:MEASview1:SYMBol:RATE?` might return 95.24E+3, indicating the calculated symbol rate is 95.24 kHz for measurement view 1.

FETCh:SQUality[:MEASview<y>]:SYMBol:RATE:ERRor? (Query only)

Returns the value of the symbol rate error in percent (%).

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

It is valid when the modulation type is 2|4|8|16FSK and `[:SENSe]:DDEMod:SYMBol:RATE:SEARch` is ON.

Group

Fetch commands

Syntax

`FETCh:SQUality[:MEASview<y>]:SYMBol:RATE:ERRor?`

Related Commands

[SENSe]:DDEMod[:SOURce<x>]:SYMBol:RATE:SEARch (on page 2047)

Returns

`<NRf>` is the symbol error percent (%).

Examples

`FETCh:SQUality:MEASview1:SYMBol:RATE:ERRor?` might return -0.002, indicating that the symbol rate error is -0.002% for measurement view 1.

FETCh:TDIagram[:MEASview<y>]:FERRor? (Query only)

Returns the frequency error in the Trellis diagram measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Trellis diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Trellis diagram).

Group

Fetch commands

Syntax

`FETCh:TDIagram[:MEASview<y>]:FERRor?`

Arguments

None

Returns

`<freq_error>::<NRf>` is the frequency error in Hz.

Examples

`FETCh:TDIagram:MEASview2:FERRor?` might return -10.7E+3, indicating the frequency error is -10.7 kHz.

FETCh:TDIagram[:MEASview<y>]:TRACe? (Query only)

Returns the Trellis diagram trace data.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Trellis diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Trellis diagram).

Group

Fetch commands

Syntax

`FETCh:TDIagram[:MEASview<y>]:TRACe?`

Arguments

None

Returns

`#<num_digit><num_byte><Y(1)><X(1)><Y(2)><X(2)>...<Y(n)><X(n)>`

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<Y(n)> and <X(n)> is the phase in degrees and the time in (symbols) pair at the n^{th} data point. The 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

FETCh:TDIagram:MEASview2:FDEVIATION? might return #3160xxxx... (160-byte data) for the Trellis diagram trace.

FETCh:TOVerview[:MEASview<y>](Query Only)

Returns the trace data in the time overview.

Conditions

Measurement views: Time overview with specified Measurement view number

Group

Fetch commands

Syntax

FETCh:TOVerview[:MEASview<y>]?

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

#<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the amplitude in dBm at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2. The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples

FETCH:TOVERVIEW:MEASVIEW1? might return #43204xxxx... (3204-byte data) for the trace in the time overview.

FETCh:TXGain:AVERage?(Query Only)

Returns the mean value of the trace.

Conditions

Measurement view: Transmission Gain

Group

Fetch commands

Syntax

FETCh:TXGain:AVERage?

Returns

<NRf> is the averaged value of the trace points in dB.

Examples

`FETCH:TXGAIN:AVERAGE?` might return 2.3, indicating that the mean value of the trace is 2.3 dB.

FETCh:TXGain:MAX? (Query Only)

Returns the maximum trace value.

Conditions

Measurement view: Transmission Gain

Group

Fetch commands

Syntax

`FETCh:TXGain:MAX?`

Returns

`<NRf>` is the maximum value of the trace in dB.

Examples

`FETCH:TXGAIN:MAX?` might return `-3.45`, indicating the maximum trace value is –3.45 dB.

FETCh:TXGain:MAXLocation? (Query Only)

Returns the frequency of the maximum trace value.

Conditions

Measurement view: Transmission Gain

Group

Fetch commands

Syntax

`FETCh:TXGain:MAXLocation?`

Returns

`<NRf>` is the frequency at the maximum trace value.

Examples

`FETCH:TXGAIN:MAXLOCATION?` might return `1E9`, indicating the maximum trace value is at 1 GHz.

FETCh:TXGain:MIN? (Query Only)

Returns the minimum trace value.

Conditions

Measurement view: Transmission Gain

Group

Fetch commands

Syntax

`FETCh:TXGain:MIN?`

Returns

`<NRf>` is the frequency at the minimum trace value in dB.

Examples

`FETCH:TXGAIN:MIN?` might return `-30`, indicating the minimum trace value is -30 dB.

FETCh:TXGain:MINLocation? (Query Only)

Returns the frequency of the minimum trace value.

Conditions

Measurement view: Transmission Gain

Group

Fetch commands

Syntax

`FETCh:TXGain:MINLocation?`

Returns

`<NRf>` is the frequency at the minimum trace value.

Examples

`FETCH:TXGAIN:MINLOCATION?` might return `1E6`, indicating the minimum trace value is at 1 MHz.

FETCh:TXGain:TRACe<x>? (Query Only)

Returns the trace data of the selected trace.

Conditions

Measurement view: Transmission Gain Where <x> is the trace 1, 2, or 3.

Group

Fetch commands

Syntax

`FETCh:TXGain:TRACe<x>?`

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the amplitude in dB at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

FETCh:TXGain:TRACe1? might return #43204xxxx... (3204 byte data) for Trace 1 in the measurement.

FETCh:WLAN:AD:CONSte:DATA:EVM? (Query Only)

Queries the maximum EVM value in data Constellation display.

Conditions

Measurement view: 802.11ad Constellation

Group

Fetch commands

Syntax

FETCh:WLAN:AD:CONSte:DATA:EVM?

Arguments

None

Returns

<NRf> = EVM value

Examples

FETCH:WLAN:AD:CONSTE:DATA:EVM? might return -42.482, indicating the EVM value present in the analysis data is -42.482 dB.

FETCh:WLAN:AD:CONSte:DATA:EVM:PEAK? (Query Only)

Queries the EVM Peak value in Constellation display.

Conditions

Measurement view: 802.11ad Constellation

Group

Fetch commands

Syntax

FETCh:WLAN:AD:CONSte:DATA:EVM:PEAK?

Arguments

None

Returns

<NRf> = EVM peak value

Examples

`FETCH:WLAN:AD:CONSTE:DATA:EVM:PEAK?` might return `-35.325`, indicating the EVM peak value is –35.325 dB.

FETCh:WLAN:AD:CONSte:DATA:EVM:PEAK:SYMBol? (Query Only)

Queries the EVM peak symbol value in Constellation display.

Conditions

Measurement view: 802.11ad Constellation

Group

Fetch commands

Syntax

`FETCh:WLAN:AD:CONSte:DATA:EVM:PEAK:SYMBol?`

Arguments

None

Returns

<NRf> = symbol value

Examples

`FETCH:WLAN:AD:CONSTE:DATA:EVM:PEAK:SYMBOL?` might return `6.312`, indicating the EVM peak symbol value is 6.312.

FETCh:WLAN:AD:CONSte:FERRor? (Query Only)

Queries the frequency Error value in Constellation display.

Conditions

Measurement view: 802.11ad Constellation

Group

Fetch commands

Syntax

`FETCh:WLAN:AD:CONSte:FERRor?`

Arguments

None

Returns

<NR3>= frequency error

Examples

`FETCH:WLAN:AD:CONSTE:FERROR?` might return `3.344+3`, indicating the frequency error value is 3.344 kHz.

FETCh:WLAN:AD:CONSte:IQDC:OFFSet? (Query Only)

Queries IQ Origin Offset value in Constellation display.

Conditions

Measurement view: 802.11ad Constellation

Group

Fetch commands

Syntax

`FETCh:WLAN:AD:CONSte:IQDC:OFFSet?`

Arguments

None

Returns

<NRf> = IQ Origin Offset value

Examples

`FETCH:WLAN:AD:CONSTE:IQDC:OFFSET?` might return `-51.841`, indicating the IQ Origin Offset value is –51.841 dB.

FETCh:WLAN:AD:EVM:TRACe? (Query Only)

Returns the EVM versus time trace data.

Conditions

Measurement views: 802.11ad EVM vs Time

Group

Fetch commands

Syntax

`FETCh:WLAN:AD:EVM:TRACe?`

Arguments

None

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the EVM value at the nth data point in percent (%),

`<data(n)>` follows 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`FETCH:WLAN:AD:EVM:TRACE?` might return `#42036xxxx...` (2036-byte data), indicating the EVM versus time trace data.

FETCh:WLAN:AD:SUMMary:AVERage:BURSt:COUNt (Query Only)

Queries the current burst number out of the total burst in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AD:SUMMary:AVERage:BURSt:COUNt?`

Arguments

None

Returns

<NR1>

Examples

`FETCH:WLAN:AD:SUMMARY:AVERAGE:BURST:COUNT?` might return `1`, indicating the current burst count is 1 out of the total burst.

FETCh:WLAN:AD:SUMMary:AVERage:BURSt:TOTAl:COUNt (Query Only)

Queries the total burst count to be averaged in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AD:SUMMary:AVERage:BURSt:TOTAl:COUNt?`

Arguments

None

Returns

<NR1>

Examples

`FETCH:WLAN:AD:SUMMARY:AVERAGE:BURST:TOTAL:COUNT?` might return `1`, indicating the total burst count to be averaged is 1.

FETCh:WLAN:AD:SUMMary:ESNR (Query Only)

Returns the value of Estimated SNR value in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Analysis Summary

Group

Fetch commands

Syntax

```
FETCh:WLAN:AD:SUMMary:ESNR?
```

Arguments

None

Returns

<NRf>

Examples

`FETCH:WLAN:AD:SUMMARY:ESNR?` might return `36.856`, indicating the estimated SNR value is 36.856 dB.

FETCh:WLAN:AD:SUMMary:EVM:DATA:AVERage (Query Only)

Queries the Avg EVM (Data) value in the 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

```
FETCh:WLAN:AD:SUMMary:EVM:DATA:AVERage?
```

Arguments

None

Returns

<NRf>

Examples

`FETCH:WLAN:AD:SUMMARY:EVM:DATA:AVERAGE?` might return `-40.7035275551`, indicating the Avg EVM (Data) value is −40.70.

FETCh:WLAN:AD:SUMMary:EVM:DATA:AVERage:STATus (Query Only)

Queries the Avg EVM (Data) status in the 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AD:SUMMary:EVM:DATA:AVERage:STATus?`

Arguments

None

Returns

<String>

PASS indicates that the test passed.

FAIL indicates that the test failed.

NA indicates that the test is not applicable.

Examples

`FETCH:WLAN:AD:SUMMARY:EVM:DATA:AVERAGE:STATUS?` might return `PASS`, indicating the Avg EVM test as passed.

FETCh:WLAN:AD:SUMMary:EVM:DATA:MAX (Query Only)

Queries the maximum EVM (Data) value in the 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AD:SUMMary:EVM:DATA:MAX?`

Arguments

None

Returns

<NRf>

Examples

`FETCH:WLAN:AD:SUMMARY:EVM:DATA:MAX?` might return `922.1968241036E-3`, indicating the maximum EVM (Data) value is 922.1968241036E-3.

FETCh:WLAN:AD:SUMMary:EVM:PACKet:CEF:EVM (Query Only)

Queries the EVM value in the CEF region of the packet in the 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax
`FETCh:WLAN:AD:SUMMary:EVM:PACKet:CEF:EVM?`

Arguments
None

Returns
<NRf>

Examples
`FETCH:WLAN:AD:SUMMARY:EVM:PACKET:CEF:EVM?` might return `6.9680400193`, indicating the EVM value in the CEF region of the packet is 6.96 dB.

FETCh:WLAN:AD:SUMMary:EVM:PACKet:CEF:PEAK (Query Only)
Queries the peak error value at the given symbol in CEF region of the packet in the 802.11ad Summary display.

Conditions
Measurement view: 802.11ad Summary

Group
Fetch commands

Syntax
`FETCh:WLAN:AD:SUMMary:EVM:PACKet:CEF:PEAK?`

Arguments
None

Returns
<NRf>

Examples
`FETCH:WLAN:AD:SUMMARY:EVM:PACKET:CEF:PEAK?` might return `13.7763872743`, indicating the peak error at the given symbol in CEF region of the packet is 13.77 dB.

FETCh:WLAN:AD:SUMMary:EVM:PACKet:CEF:SYMBol (Query Only)
Queries the symbol at which the peak error occurred in the CEF region of the packet in the 802.11ad Summary display.

Conditions
Measurement view: 802.11ad Summary

Group
Fetch commands

Syntax
`FETCh:WLAN:AD:SUMMary:EVM:PACKet:CEF:SYMBol?`

Arguments

None

Returns

<NRf>

Examples

`FETCH:WLAN:AD:SUMMARY:EVM:PACKET:CEF:SYMBOL?` might return `7.4020000000E+3`, indicating the peak error occurred in symbol 7.40 in the CEF region of the packet.

FETCh:WLAN:AD:SUMMary:EVM:PACKet:DATA:EVM (Query Only)

Queries the EVM value in the DATA region of the packet in the 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AD:SUMMary:EVM:PACKet:DATA:EVM?`

Arguments

None

Returns

<NRf>

Examples

`FETCH:WLAN:AD:SUMMARY:EVM:PACKET:DATA:EVM?` might return `-33.5668742100`, indicating the EVM value in the DATA region of the packet is -33.56 dB.

FETCh:WLAN:AD:SUMMary:EVM:PACKet:DATA:PEAK (Query Only)

Queries the peak error value at the given symbol in the DATA region of the packet in the 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AD:SUMMary:EVM:PACKet:DATA:PEAK?`

Arguments

None

Returns

<NRf>

Examples

`FETCH:WLAN:AD:SUMMARY:EVM:PACKET:DATA:PEAK?` might return `-26.5964736883`, indicating the peak error value is −26.59 dB at the given symbol in DATA region of the packet.

FETCh:WLAN:AD:SUMMary:EVM:PACKet:DATA:STATus (Query Only)

Queries the status of the DATA packet in Summary display in the 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AD:SUMMary:EVM:PACKet:DATA:STATus?`

Arguments

None

Returns

<String>

PASS indicates that the test passed.

FAIL indicates that the test failed.

NA indicates that the test is not Applicable.

Examples

`FETCH:WLAN:AD:SUMMARY:EVM:PACKET:DATA:STATUS?` might return `PASS`, indicating the DATA packet test as passed.

FETCh:WLAN:AD:SUMMary:EVM:PACKet:DATA:SYMBol (Query Only)

Queries the symbol at which the peak error occurred in the DATA region of the packet in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AD:SUMMary:EVM:PACKet:DATA:SYMBol?`

Arguments

None

Returns

<NRf>

Examples

`FETCH:WLAN:AD:SUMMARY:EVM:PACKET:DATA:SYMBOL?` might return `4.6610000000E+3`, indicating that the symbol peak error occurred at 4.66 in the DATA region of the packet.

FETCh:WLAN:AD:SUMMary:EVM:PACKet:GUARd:EVM (Query Only)

Queries the EVM value in Guard region of the Packet in the 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AD:SUMMary:EVM:PACKet:GUARd:EVM?`

Arguments

None

Returns

<NRf>

Examples

`FETCH:WLAN:AD:SUMMARY:EVM:PACKET:GUARD:EVM` might return `-65.4983500315`, indicating the EVM value in the Guard region of the packet is –65.49 dB.

FETCh:WLAN:AD:SUMMary:EVM:PACKet:GUARd:PEAK (Query Only)

Queries the peak error value at the given symbol in the Guard region of the packet in the 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AD:SUMMary:EVM:PACKet:GUARd:PEAK?`

Arguments

None

Returns

<NRf>

Examples

FETCH:WLAN:AD:SUMMARY:EVM:PACKET:GUARD:PEAK? might return -26.5964736883, indicating the peak error at the given symbol in the Guard region of the packet is –26.59 dB.

FETCh:WLAN:AD:SUMMary:EVM:PACKet:GUARd:SYMBol (Query Only)

Queries the symbol at which the peak error occurred in the Guard region of the packet in the 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

FETCh:WLAN:AD:SUMMary:EVM:PACKet:GUARd:SYMBol?

Arguments

None

Returns

<NRf>

Examples

FETCH:WLAN:AD:SUMMARY:EVM:PACKET:GUARD:SYMBOL? might return 3.3340000000E+3, indicating the peak error occurred at symbol 3.33 in the Guard region of the packet.

FETCh:WLAN:AD:SUMMary:EVM:PACKet:HEADer:EVM (Query Only)

Queries the EVM value in the Header region of the packet in the 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

FETCh:WLAN:AD:SUMMary:EVM:PACKet:HEADer:EVM?

Arguments

None

Returns

<NRf>

Examples

FETCH:WLAN:AD:SUMMARY:EVM:PACKET:HEADER:EVM? might return -67.1981081169, indicating the EVM value in the Header region of the packet is –67.19 dB.

FETCh:WLAN:AD:SUMMary:EVM:PACKet:HEADer:PEAK (Query Only)

Queries the peak error at the given symbol in the Header region if the packet in the 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AD:SUMMary:EVM:PACKet:HEADer:PEAK?`

Arguments

None

Returns

<NRf>

Examples

`FETCH:WLAN:AD:SUMMARY:EVM:PACKET:HEADER:PEAK?` might return `-60.9603086224`, indicating the peak error at the given symbol is −60.96 dB in the Header region of the packet.

FETCh:WLAN:AD:SUMMary:EVM:PACKet:HEADer:SYMBol (Query Only)

Queries the symbol at which the peak Error occurred in the Header region of the packet in the 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AD:SUMMary:EVM:PACKet:HEADer:SYMBol?`

Arguments

None

Returns

<NRf>

Examples

`FETCH:WLAN:AD:SUMMARY:EVM:PACKET:HEADER:SYMBOL?` might return `3.3970000000E+3`, indicating the symbol peak error occurred is 3.397 in Header region of the packet.

FETCh:WLAN:AD:SUMMary:EVM:PACKet:STF:EVM (Query Only)

Queries the EVM value in the STF region of the packet in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AD:SUMMary:EVM:PACKet:STF:EVM?`

Arguments

None

Returns

<NRf>

Examples

`FETCH:WLAN:AD:SUMMARY:EVM:PACKET:STF:EVM?` might return `-22.9729890175`, indicating the EVM value in the STF region of the packet is −22.97 dB.

FETCh:WLAN:AD:SUMMary:EVM:PACKet:STF:PEAK (Query Only)

Queries the peak error value at the given symbol in the STF region of the packet in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AD:SUMMary:EVM:PACKet:STF:PEAK?`

Arguments

None

Returns

<NRf>

Examples

`FETCH:WLAN:AD:SUMMARY:EVM:PACKET:STF:PEAK?` might return `38.8123661280`, indicating the peak error at the given symbol in the STF region of the packet is 38.81 dB.

FETCh:WLAN:AD:SUMMary:EVM:PACKet:STF:SYMBol (Query Only)

Queries the symbol at which the peak error occurred in the STF region of the packet in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AD:SUMMary:EVM:PACKet:STF:SYMBol?`

Arguments

None

Returns

<NRf>

Examples

`FETCH:WLAN:AD:SUMMARY:EVM:PACKET:STF:SYMBOL?` might return `0.0`, indicating the peak error occurred at the 0.0 symbol in the STF region of the packet.

FETCh:WLAN:AD:SUMMary:FERRor:AVERage (Query Only)

Queries the Avg Frequency Error value in the 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AD:SUMMary:FERRor:AVERage?`

Arguments

None

Returns

<NRf>

Examples

`FETCH:WLAN:AD:SUMMARY:FERROR:AVERAGE?` might return `-114.1332879317E-3`, indicating the Avg Frequency error is -114.1332879317E-3 ppm.

FETCh:WLAN:AD:SUMMary:FERRor:AVERage:STATus (Query Only)

Queries the Avg Frequency error status in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AD:SUMMary:FERRor:AVERage:STATus?`

Arguments

None

Returns

<String>

PASS indicates that the test passed.

FAIL indicates that the test failed.

NA indicates that the test is not applicable.

Examples

`FETCH:WLAN:AD:SUMMARY:FERROR:AVERAGE:STATUS?` might return `FAIL`, indicating the Avg Frequency error test failed.

FETCh:WLAN:AD:SUMMary:FERRor:HERTz (Query Only)

Queries the Frequency Error value in 802.11ad Summary display in Hz.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AD:SUMMary:FERRor:HERTz?`

Arguments

None

Returns

<NRf>

Examples

`FETCH:WLAN:AD:SUMMARY:FERROR:HERTZ?` might return `2.875E+3`, indicating the Frequency Error value is 2.875E+3 Hz.

FETCh:WLAN:AD:SUMMary:FERRor:MAX (Query Only)

Queries the Max Frequency error value in the 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax
`FETCh:WLAN:AD:SUMMary:FERRor:MAX?`

Arguments
None

Returns
<NRf>

Examples
`FETCH:WLAN:AD:SUMMARY:FERROR:MAX?` might return `-114.1332879317E-3`, indicating the max frequency error is -114.1332879317E-3 ppm.

FETCh:WLAN:AD:SUMMary:FERRor:PPM (Query Only)
Queries the Frequency Error value in 802.11ad Summary display in ppm.

Conditions
Measurement view: 802.11ad Summary

Group
Fetch commands

Syntax
`FETCh:WLAN:AD:SUMMary:FERRor:PPM?`

Arguments
None

Returns
<NRf>

Examples
`FETCH:WLAN:AD:SUMMARY:FERROR:PPM?` might return `575.023E-3`, indicating the Frequency Error value is 575.023E-3 ppm.

FETCh:WLAN:AD:SUMMary:FERRor:STATus (Query Only)
Queries the Frequency error status in the 802.11ad Summary display.

Conditions
Measurement view: 802.11ad Summary

Group
Fetch commands

Syntax
`FETCh:WLAN:AD:SUMMary:FERRor:STATus?`

Arguments

None

Returns

<String>

PASS indicates that the test passed.

FAIL indicates that the test failed.

NA indicates that the test is not applicable.

Examples

`FETCH:WLAN:AD:SUMMARY:FERROR:STATUS?` might return `PASS`, indicating the frequency error test passed.

FETCh:WLAN:AD:SUMMary:IQ:GAIN:IMBALance (Query Only)

Queries IQ Gain Imbalance value in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AD:SUMMary:IQ:GAIN:IMBALance?`

Arguments

None

Returns

<NRf>

Examples

`FETCH:WLAN:AD:SUMMARY:IQ:GAIN:IMBALANCE?` might return `69.2700818181E-3`, indicating IQ Gain Imbalance value is 69.2700818181E-3% .

FETCh:WLAN:AD:SUMMary:IQ:ORIGin:OFFSet (Query Only)

Queries IQ Origin Offset value in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AD:SUMMary:IQ:ORIGin:OFFSet?`

Arguments

None

Returns

<NRf>

Examples

`FETCH:WLAN:AD:SUMMARY:IQ:ORIGIN:OFFSET?` might return `-160.0000000000`, indicating the IQ Origin Offset value is −160.0 dB.

FETCh:WLAN:AD:SUMMary:IQ:QUADrature:ERROr (Query Only)

Queries IQ Quadrature Error value in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AD:SUMMary:IQ:QUADrature:ERROr?`

Arguments

None

Returns

<NRf>

Examples

`FETCH:WLAN:AD:SUMMARY:IQ:QUADRATURE:ERROR?` might return `-220.5967903137E-3`, indicating IQ Quadrature Error value is -220.5967903137E-3 degrees.

FETCh:WLAN:AD:SUMMary:PACKet:AGGRegation (Query Only)

Queries the value of Aggregation packet information in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AD:SUMMary:PACKet:AGGRegation?`

Arguments

None

Returns

<Boolean> 0,1

Examples

FETCH:WLAN:AD:SUMMARY:PACKET:AGGREGATION? might return "0", indicating the Aggregation packet value is 0.

FETCh:WLAN:AD:SUMMary:PACKet:BTR (Query Only)

Queries the value of Beam Tracking Request packet information in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

FETCh:WLAN:AD:SUMMary:PACKet:BTR?

Arguments

None

Returns

<Boolean> 0,1

Examples

FETCH:WLAN:AD:SUMMARY:PACKET:BTR? might return 0, indicating the Beam Tracking Request packet value is 0.

FETCh:WLAN:AD:SUMMary:PACKet:HCS (Query Only)

Queries the value of HCS packet information in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

FETCh:WLAN:AD:SUMMary:PACKet:HCS?

Arguments

None

Returns

<String>

Examples

FETCH:WLAN:AD:SUMMARY:PACKET:HCS? might return "1001010000001000", indicating the HCS packet value is 1001010000001000.

FETCh:WLAN:AD:SUMMary:PACKet:LENGth (Query Only)

Queries the value of Length packet information in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

```
FETCh:WLAN:AD:SUMMary:PACKet:LENGth?
```

Arguments

None

Returns

<String>

Examples

`FETCH:WLAN:AD:SUMMARY:PACKET:LENGTH?` might return `"1000"`, indicating the Length packet value is 1000.

FETCh:WLAN:AD:SUMMary:PACKet:MCS (Query Only)

Queries the value of MCS packet information in Summary display in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

```
FETCh:WLAN:AD:SUMMary:PACKet:MCS?
```

Arguments

None

Returns

<NR1> Range is 0 to 12

Examples

`FETCH:WLAN:AD:SUMMARY:PACKET:MCS?` might return `5`, indicating the MCS packet value is 5.

FETCh:WLAN:AD:SUMMary:PACKet:PPDU (Query Only)

Queries the value of Additional PPDU packet information in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

```
FETCh:WLAN:AD:SUMMary:PACKet:PPDU?
```

Arguments

None

Returns

<Boolean> 0,1

Examples

`FETCH:WLAN:AD:SUMMARY:PACKET:PPDU?` might return `"0"`, indicating the Additional PPDU packet value is 0.

FETCh:WLAN:AD:SUMMary:PACKet:RSSI (Query Only)

Queries the value of Last RSSI packet information in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

```
FETCh:WLAN:AD:SUMMary:PACKet:RSSI?
```

Arguments

None

Returns

<NRf>

Examples

`FETCH:WLAN:AD:SUMMARY:PACKET:RSSI?` might return `"0000"`, indicating the Last RSSI packet value is 0000.

FETCh:WLAN:AD:SUMMary:PACKet:SCRAmbler:INITialization (Query Only)

Queries the value of Scrambler initialization packet information in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

FETCh:WLAN:AD:SUMMary:PACKet:SCRAmbler:INITialization?

Arguments

None

Returns

<String>

Examples

FETCH:WLAN:AD:SUMMARY:PACKET:SCRAMBLER:INITIALIZATION? might return "0101110", indicating the Scrambler Initialization packet value is 0101110.

FETCh:WLAN:AD:SUMMary:PACKet:TAROUND (Query Only)

Queries the value of the Turnaround packet information in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

FETCh:WLAN:AD:SUMMary:PACKet:TAROUND?

Arguments

None

Returns

<Boolean> 0,1

Examples

FETCH:WLAN:AD:SUMMARY:PACKET:TAROUND? might return "1", indicating the Turnaround packet value is 1.

FETCh:WLAN:AD:SUMMary:PACKet:TLEN (Query Only)

Queries the value of Training Length packet information in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

FETCh:WLAN:AD:SUMMary:PACKet:TLEN?

Arguments

None

Returns

<NRf>

Examples

`FETCH:WLAN:AD:SUMMARY:PACKET:TLEN?` might return `21`, indicating the Training Length packet value is 21.

FETCh:WLAN:AD:SUMMary:PACKet:TYPE (Query Only)

Queries the value of Packet Type information in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AD:SUMMary:PACKet:TYPE?`

Arguments

None

Returns

<Boolean> 0,1

Examples

`FETCH:WLAN:AD:SUMMARY:PACKET:TYPE?` might return `"0"`, indicating the Packet Type value is 0.

FETCh:WLAN:AD:SUMMary:RCPI (Query Only)

Queries the RCPI value in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AD:SUMMary:RCPI?`

Arguments

None

Returns

<NRf>

Examples

`FETCH:WLAN:AD:SUMMARY:RCPI?` might return `207.0000000000`, indicating the RCPI is 207.

FETCh:WLAN:AD:SUMMary:RF:POWer (Query Only)

Queries the RF Output Power value in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Fetch commands

Syntax

```
FETCh:WLAN:AD:SUMMary:RF:POWer?
```

Arguments

None

Returns

<NRf>

Examples

`FETCH:WLAN:AD:SUMMARY:RF:POWER?` might return `-6.3434391022`, indicating the RF Output Power is -6.3434391022 dBm.

FETCh:WLAN:AY:CONSte:DATA:EVM:PEAK? (Query Only)

Query the Data Peak Error value in the Constellation display.

Conditions

Measurement view: 802.11ay Analysis Constellation

Group

Fetch commands

Syntax

```
FETCh:WLAN:AY:CONSte:DATA:EVM:PEAK?
```

Returns

```
<NRf>
```

You must select the units in the EVM tab.

Examples

`FETC:WLAN:AY:CONS:DATA:EVM:PEAK?` may return `-35.325`, which means -35.325 is the Data Peak Error value.

FETCh:WLAN:AY:CONSte:DATA:EVM:PEAK:SYMBol? (Query Only)

Query the Symbol position that contains the maximum Data EVM value in the Constellation display.

Conditions

Measurement view: 802.11ay Analysis Constellation

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:CONSte:DATA:EVM:PEAK:SYMBol?`

Returns

`<NRf>`

Examples

`FETC:WLAN:AY:CONS:DATA:EVM:PEAK:SYMB?` may return `6.312e+0003`, which indicates that the 6312 Symbol contains the maximum Data EVM value.

FETCh:WLAN:AY:CONSte:DATA:EVM? (Query Only)

Query the maximum EVM value present in the analysis data in the Constellation display.

Conditions

Measurement view: 802.11ay Analysis Constellation

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:CONSte:DATA:EVM?`

Returns

`<NRf>`

You must select the units in the EVM tab.

Examples

`FETC:WLAN:AY:CONS:DATA:EVM?` may return `-42.482`, which means the maximum EVM value in the analysis data is -42.482.

FETCh:WLAN:AY:CONSte:FERRor? (Query Only)

Query the Frequency Error value in the Constellation display.

Conditions

Measurement view: 802.11ay Analysis Constellation

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:CONSte:FERRor?`

Returns

`<NRf>`

Units: Hz

Examples

`FETC:WLAN:AY:CONS:FERR?` may return `0.0000`, which indicates that the Frequency Error value is 0.0000 Hz.

FETCh:WLAN:AY:CONSte:FREQuency:ERROr? (Query Only)

Query the Frequency Error value in the Constellation display.

Conditions

Measurement view: 802.11ay Analysis Constellation

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:CONSte:FREQuency:ERROr?`

Returns

`<NRf>`

Units: Hz

Examples

`FETC:WLAN:AY:CONS:FREQ:ERRO?` may return `0.0000`, which means the Frequency Error value is 0.0000 Hz.

FETCh:WLAN:AY:CONSte:IQDC:OFFSet? (Query Only)

Query the IQ Origin Offset value in the Constellation display.

Conditions

Measurement view: 802.11ay Analysis Constellation

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:CONSte:IQDC:OFFSet?`

Returns

`<NRf>`

Units: dB

Examples

`FETC:WLAN:AY:CONS:IQDC:OFFS?` may return `-51.841`, which indicates that the IQ Origin Offset value is -51.841 dB.

FETCh:WLAN:AY:EVM:TRACe? (Query Only)

Query the EVM versus Time trace data after doing a single run.

Conditions

Measurement views: 802.11ay Analysis EVM

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:EVM:TRACe?`

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

where:

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the EVM value at the *n*th data point in percentage (%).

`<data(n)>` follows 4-byte little endian floating point format specified in IEEE 488.2.

Examples

`FETC:WLAN:AY:EVM:TRAC?` may return `#42036xxxx...` (2036-byte data) for the EVM versus time trace.

FETCh:WLAN:AY:SUMMary:AVERage:BURSt:COUNt? (Query Only)

Query the current Burst Count in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:SUMMary:AVERage:BURSt:COUNt?`

Returns

`<NR1>`

Examples

`FETC:WLAN:AY:SUMM:AVER:BURS:COUN?` may return `1`, which indicates that the current Burst Count is 1.

FETCh:WLAN:AY:SUMMary:AVERage:BURSt:TOTAl:COUNt? (Query Only)

Query the Total Burst Count to be averaged.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

FETCh:WLAN:AY:SUMMary:AVERage:BURSt:TOTAl:COUNt?

Returns

<NR1>

Examples

FETC:WLAN:AY:SUMM:AVER:BURS:TOTA:COUN? may return 1, which indicates that the Total Burst Count to be averaged is 1.

FETCh:WLAN:AY:SUMMary:ESNR? (Query Only)

Query the Estimated SNR (Signal-to-Noise Ratio) value in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

FETCh:WLAN:AY:SUMMary:ESNR?

Returns

<NRf>

Units: dB

Examples

FETC:WLAN:AY:SUMM:ESNR? may return 34.50, which indicates that the Estimated SNR value is 34.50 dB.

FETCh:WLAN:AY:SUMMary:EVM:DATA:AVERage? (Query Only)

Query the Average EVM (Data) value in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

FETCh:WLAN:AY:SUMMary:EVM:DATA:AVERage?

Returns

<NRf>

Units: dB or % (based on the selected EVM unit)

Examples

FETC:WLAN:AY:SUMM:EVM:DATA:AVER? may return -40.7035275551, which indicates that the Average EVM (Data) value is -40.7035275551%.

FETCh:WLAN:AY:SUMMary:EVM:DATA:AVERage:STATus? (Query Only)

Query the Average EVM (Data) Status in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:SUMMary:EVM:DATA:AVERage:STATus?`

Returns

<string>

Examples

`FETC:WLAN:AY:SUMM:EVM:DATA:AVER:STAT?` may return `PASS`, `FAIL`, or `NA`.

FETCh:WLAN:AY:SUMMary:EVM:DATA:MAX? (Query Only)

Query the Maximum EVM (Data) value in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:SUMMary:EVM:DATA:MAX?`

Returns

`<NRf>`

Units: dB or % (based on the selected EVM unit)

Examples

`FETC:WLAN:AY:SUMM:EVM:DATA:MAX?` may return `922.1968241036e-3`, which indicates that the Maximum EVM (Data) value is 922.1968241036e-3 dB.

FETCh:WLAN:AY:SUMMary:EVM:DATA:SIGma? (Query Only)

Query the Standard Deviation (1-Sigma) of the EVM Data.

When the EVM unit is in dB, it queries the Standard Deviation after converting evm values to the dB scale.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax
`FETCh:WLAN:AY:SUMMary:EVM:DATA:SIGma?`

Returns
`<NRf>`

Units: dB

Examples
`FETC:WLAN:AY:SUMM:EVM:DATA:SIG?` may return `1.8950`, which indicates that the 1-Sigma EVM Data value is 1.8950 dB.

FETCh:WLAN:AY:SUMMary:EVM:PACKet:DATA:EVM? (Query Only)
Query the EVM value that occurs in the Data Region of the packet in the Summary display.

Conditions
Measurement view: 802.11ay Analysis Summary

Group
Fetch commands

Syntax
`FETCh:WLAN:AY:SUMMary:EVM:PACKet:DATA:EVM?`

Returns
`<NRf>`

Units: dB or % (based on the EVM unit selected)

Examples
`FETC:WLAN:AY:SUMM:EVM:PACK:DATA:EVM?` may return `-33.5668742100`, which indicates that the EVM value is -33.5668742100%.

FETCh:WLAN:AY:SUMMary:EVM:PACKet:DATA:PEAK? (Query Only)
Query the value of the Peak Error that occurs at the given Symbol in the Data Region of the packet in the Summary display.

Conditions
Measurement view: 802.11ay Analysis Summary

Group
Fetch commands

Syntax
`FETCh:WLAN:AY:SUMMary:EVM:PACKet:DATA:PEAK?`

Returns
`<NRf>`

Units: dB or % (based on the EVM unit selected)

Examples

`FETC:WLAN:AY:SUMM:EVM:PACK:DATA:PEAK?` may return `-26.5964736883`, which indicates that the value of the Peak Error is -26.5964736883 at the given Symbol.

FETCh:WLAN:AY:SUMMary:EVM:PACKet:DATA:STATus? (Query Only)

Query the status of the Data Region of the packet in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:SUMMary:EVM:PACKet:DATA:STATus?`

Returns

`<string>`

Examples

`FETC:WLAN:AY:SUMM:EVM:PACK:DATA:STAT?` may return `PASS`, `FAIL`, or `NA`.

FETCh:WLAN:AY:SUMMary:EVM:PACKet:DATA:SYMBol? (Query Only)

Query the Symbol at which the Peak Error occurs in the Data Region of the packet in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:SUMMary:EVM:PACKet:DATA:SYMBol?`

Returns

`<NRf>`

Examples

`FETC:WLAN:AY:SUMM:EVM:PACK:DATA:SYMB?` may return `4.6610000000e+3`, which indicates that the Peak Error occurs at the 4.6610000000e+3 Symbol.

FETCh:WLAN:AY:SUMMary:EVM:PACKet:EDMG:CEF:EVM? (Query Only)

Query the EVM value in the 11ay 2.16 GHz BW and EDMG STF for 11ay 4.32 GHz BW of the packet in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:SUMMary:EVM:PACKet:EDMG:CEF:EVM?`

Returns

`<NRf>`

Units: dB or % (based on the EVM unit selected)

Examples

`FETC:WLAN:AY:SUMM:EVM:PACK:EDMG:CEF:EVM?` may return `6.9680400193`, which indicates that the EVM value is -6.9680400193 dB.

FETCh:WLAN:AY:SUMMary:EVM:PACKet:EDMG:CEF:PEAK? (Query Only)

Query the Peak Error that occurs at the given Symbol in the 11ay 2.16 GHz BW and EDMG CEF for 11ay 4.32 GHz BW of the packet in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:SUMMary:EVM:PACKet:EDMG:CEF:PEAK?`

Returns

<NRf>

Units: dB or % (based on the EVM unit selected)

Examples

`FETC:WLAN:AY:SUMM:EVM:PACK:EDMG:CEF:PEAK?` may return `13.7763872743`, which indicates that the Peak Error value is 13.7763872743% at the given Symbol.

FETCh:WLAN:AY:SUMMary:EVM:PACKet:EDMG:CEF:SYMBol? (Query Only)

Query the Symbol at which the Peak Error occurs in the 11ay 2.16 GHz BW and EDMG CEF for 11ay 4.32 GHz BW of the packet in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:SUMMary:EVM:PACKet:EDMG:CEF:SYMBol?`

Returns

`<NRf>`

Examples

`FETC:WLAN:AY:SUMM:EVM:PACK:EDMG:CEF:SYMB?` may return `7.4020000000e+3`, which indicates that the Peak Error occurs at the 7.4020000000e+3 Symbol.

FETCh:WLAN:AY:SUMMary:EVM:PACKet:EDMG:STF:EVM? (Query Only)

Query the EVM value in the 11ay 2.16 GHz BW and EDMG STF for 11ay 4.32 GHz BW of the packet in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:SUMMary:EVM:PACKet:EDMG:STF:EVM?`

Returns

`<NRf>`

Units: dB or % (based on the EVM unit selected)

Examples

`FETC:WLAN:AY:SUMM:EVM:PACK:EDMG:STF:EVM?` may return `-22.9729890175`, which indicates that the EVM value is -22.9729890175 dB.

FETCh:WLAN:AY:SUMMary:EVM:PACKet:EDMG:STF:PEAK? (Query Only)

Query the Peak Error that occurs at the given Symbol in the 11ay 2.16 GHz BW and EDMG STF for 11ay 4.32 GHz BW of the packet in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:SUMMary:EVM:PACKet:EDMG:STF:PEAK?`

Returns

`<NRf>`

Units: dB or % (based on the EVM unit selected)

Examples

`FETC:WLAN:AY:SUMM:EVM:PACK:EDMG:STF:PEAK?` may return `38.8123661280`, which indicates that the Peak Error value is 38.8123661280% at the given Symbol.

FETCh:WLAN:AY:SUMMary:EVM:PACKet:EDMG:STF:SYMBol? (Query Only)

Query the Symbol at which the Peak Error occurs in the 11ay 2.16 GHz BW and EDMG STF for 11ay 4.32 GHz BW of the packet in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:SUMMary:EVM:PACKet:EDMG:STF:SYMBol?`

Returns

`<NRf>`

Examples

`FETC:WLAN:AY:SUMM:EVM:PACK:EDMG:STF:SYMB?` may return `0.0000`, which indicates that the Peak Error occurs at the 0.0000 Symbol.

FETCh:WLAN:AY:SUMMary:EVM:PACKet:GUARd:EVM? (Query Only)

Query the EVM value in the Guard Region of the packet in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax
FETCh:WLAN:AY:SUMMary:EVM:PACKet:GUARd:EVM?

Returns
<NRf>

Units: dB or % (based on the EVM unit selected)

Examples
FETC:WLAN:AY:SUMM:EVM:PACK:GUAR:EVM? may return -65.4983500315, which indicates that the value of the EVM is -65.4983500315 dB.

FETCh:WLAN:AY:SUMMary:EVM:PACKet:GUARd:PEAK? (Query Only)
Query the Peak Error that occurs at the given Symbol in the Guard Region of the packet in the Summary display.

Conditions
Measurement view: 802.11ay Analysis Summary

Group
Fetch commands

Syntax
FETCh:WLAN:AY:SUMMary:EVM:PACKet:GUARd:PEAK?

Returns
<NRf>

Units: dB or % (based on the EVM unit selected)

Examples
FETC:WLAN:AY:SUMM:EVM:PACK:GUAR:PEAK? may return -26.5964736883, which indicates that the value of the Peak Error is -26.5964736883 at the given Symbol.

FETCh:WLAN:AY:SUMMary:EVM:PACKet:GUARd:SYMBol? (Query Only)
Query the Symbol at which the Peak Error occurs in the Guard Region of the packet in the Summary display.

Conditions
Measurement view: 802.11ay Analysis Summary

Group
Fetch commands

Syntax
FETCh:WLAN:AY:SUMMary:EVM:PACKet:GUARd:SYMBol?

Returns
<NRf>

Units: dB or % (based on the EVM unit selected)

Examples

`FETC:WLAN:AY:SUMM:EVM:PACK:GUAR:SYMB?` may return `3.3340000000e+3`, which indicates that the Peak Error occurs at the 3.3340000000e+3 Symbol.

FETCh:WLAN:AY:SUMMary:EVM:PACKet:HEADer:EVM? (Query Only)

Query the EVM value in the Header Region of the packet (excluding GI symbols) in the Summary display. This command is not applicable for 11ay 4.32 GHz BW and the value can be ignored.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:SUMMary:EVM:PACKet:HEADer:EVM?`

Returns

`<NRf>`

Units: dB or % (based on the EVM unit selected)

Examples

`FETC:WLAN:AY:SUMM:EVM:PACK:HEAD:EVM?` may return `-67.1981081169`, which indicates that the EVM value in the Header Region is -67.198108116 dB.

FETCh:WLAN:AY:SUMMary:EVM:PACKet:HEADer:PEAK? (Query Only)

Query the Peak Error that occurs at the given Symbol in Header Region (excluding GI symbols) of the packet in the Summary display. This command is not applicable for 11ay 4.32 GHz BW and the value can be ignored.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:SUMMary:EVM:PACKet:HEADer:PEAK?`

Returns

`<NRf>`

Units: dB or % (based on the EVM unit selected)

Examples

FETC:WLAN:AY:SUMM:EVM:PACK:HEAD:PEAK? may return -60.9603086224, which indicates that the Peak Error value is -60.9603086224 dB.

FETCh:WLAN:AY:SUMMary:EVM:PACKet:HEADer:SYMBol? (Query Only)

Query the Symbol at which the Peak Error occurs in the Header Region of the packet (excluding GI symbols) in the Summary display. This command is not applicable for 11ay 4.32 GHz BW and the value can be ignored.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

FETCh:WLAN:AY:SUMMary:EVM:PACKet:HEADer:SYMBol?

Returns

<NRf>

Units: dB or % (based on the EVM unit selected)

Examples

FETC:WLAN:AY:SUMM:EVM:PACK:HEAD:SYMB? may return 3.3970000000e+3, which indicates that the peak error occurs at the 3.3970000000e+3 Symbol.

FETCh:WLAN:AY:SUMMary:EVM:PACKet:PRE1Edmg:A1Header:EVM? (Query

Query the value of the Pre-EDGM1 Header-A EVM packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

FETCh:WLAN:AY:SUMMary:EVM:PACKet:PRE1Edmg:A1Header:EVM?

Returns

<string>

Units: dB or % (based on the EVM unit selected)

Examples

FETC:WLAN:AY:SUMM:EVM:PACK:PRE1E:A1H:EVM? may return -34.3702, which indicates that the value of the Pre-EDGM1 Header-A EVM packet is -34.3702 dB.

FETCh:WLAN:AY:SUMMary:EVM:PACKet:PRE1Edmg: CEF1:EVM? (Query Only)

Query the value of the Pre-EDMG1 CEF EVM packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:SUMMary:EVM:PACKet:PRE1Edmg:CEF1:EVM?`

Returns

`<string>`

Units: dB or % (based on the EVM unit selected)

Examples

`FETC:WLAN:AY:SUMM:EVM:PACK:PRE1E:CEF1:EVM?` may return `-34.5666`, which indicates that the value of the Pre-EDMG1 CEF EVM packet is -34.5666 dB.

FETCh:WLAN:AY:SUMMary:EVM:PACKet:PRE1Edmg: GUARd1:EVM? (Query

Query the value of the Pre-EDMG1 EVM Guard region of the packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:SUMMary:EVM:PACKet:PRE1Edmg:GUARd1:EVM?`

Returns

`<string>`

Units: dB or % (based on the EVM unit selected)

Examples

`FETC:WLAN:AY:SUMM:EVM:PACK:PRE1ELGUAR1:EVM?` may return `-34.1327`, which indicates that the value of the Pre-EDMG1 EVM Guard Region is -34.1327 dB.

FETCh:WLAN:AY:SUMMary:EVM:PACKet:PRE1Edmg :HEADer1:EVM? (Query

Query the value of the Pre-EDMG1 L-Header EVM packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:SUMMary:EVM:PACKet:PRE1Edmg:HEADer1:EVM?`

Returns

`<string>`

Units: dB or % (based on the EVM unit selected)

Examples

`FETC:WLAN:AY:SUMM:EVMPACK:PRE1E:HEAD1:EVM?` may return `-34.4326`, which indicates that the value of the Pre-EDMG1 L-Header EVM packet is -34.4326 dB.

FETCh:WLAN:AY:SUMMary:EVM:PACKet:PRE1Edmg:STF1:EVM? (Query Only)

Query the value of the Pre-EDMG1 STF EVM packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:SUMMary:EVM:PACKet:PRE1Edmg:STF1:EVM?`

Returns

`<string>`

Units: dB or % (based on the EVM unit selected)

Examples

`FETC:WLAN:AY:SUMM:EVM:PACK:PRE1E:STF1:EVM?` may return `-33.6325`, which indicates that the value of the Pre-EDMG1 STF1 EVM packet is -33.6325 dB.

FETCh:WLAN:AY:SUMMary:EVM:PACKet:PRE2Edmg:A2Header:EVM? (Query

Query the value of the Pre-EDMG2 EDMG Header-A EVM packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax
`FETCh:WLAN:AY:SUMMary:EVM:PACKet:PRE2Edmg:A2Header:EVM?`

Returns
`<string>`

Examples
`FETC:WLAN:AY:SUMM:EVM:PACK:PRE2E:A2H:EVM?` may return `-34.37`, which indicates that the value of the Pre-EDMG2 EDMG-Header-A EVM packet is -34.37 dB.

FETCh:WLAN:AY:SUMMary:EVM:PACKet:PRE2Edmg:CEF2:EVM? (Query Only)
Query the value of the Pre-EDMG2 CEF EVM packet information in the Summary display.

Conditions
Measurement view: 802.11ay Analysis Summary

Group
Fetch commands

Syntax
`FETCh:WLAN:AY:SUMMary:EVM:PACKet:PRE2Edmg:CEF2:EVM?`

Returns
`<string>`

Units: dB or % (based on the EVM unit selected)

Examples
`FETC:WLAN:AY:SUMM:EVM:PACK:PRE2E:CEF2:EVM?` may return `-34.5665`, which indicates that the value of the Pre-EDMG2 CEF EVM packet is -34.5665 dB.

FETCh:WLAN:AY:SUMMary:EVM:PACKet:PRE2Edmg:GUARd2:EVM? (Query
Query the value of the Pre-EDMG2 Guard EVM packet information in the Summary display.

Conditions
Measurement view: 802.11ay Analysis Summary

Group
Fetch commands

Syntax
`FETCh:WLAN:AY:SUMMary:EVM:PACKet:PRE2Edmg:GUARd2:EVM?`

Returns
`<string>`

Units: dB or % (based on the EVM unit selected)

Examples

`FETC:WLAN:AY:SUMM:EVM:PACK:PRE2E:GUAR:EVM?` may return `-33.137`, which indicates that the value of the Pre-EDMG2 Guard EVM packet is -33.137 dB.

FETCh:WLAN:AY:SUMMary:EVM:PACKet:PRE2Edmg: HEADer2:EVM? (Query

Query the value of the Pre-EDMG2 L-Header EVM packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:SUMMary:EVM:PACKet:PRE2Edmg:HEADer2:EVM?`

Returns

`<string>`

Units: dB or % (based on the EVM unit selected)

Examples

`FETC:WLAN:AY:SUMM:EVM:PACK:PRE2E:HEAD2:EVM?` may return `-34.4321`, which indicates that the value of the Pre-EDMG2 L-Header EVM packet is -34.4321 dB.

FETCh:WLAN:AY:SUMMary:EVM:PACKet:PRE2Edmg: STF2:EVM? (Query Only)

Query the value of the Pre-EDMG2 STF EVM packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:SUMMary:EVM:PACKet:PRE2Edmg:STF2:EVM?`

Returns

<string>

Units: dB or % (based on the EVM unit selected)

Examples

`FETC:WLAN:AY:SUMM:EVM:PACK:PRE2E:STF2:EVM?` may return `-33.6324`, which indicates that the value of the Pre-EDMG2 STF2 EVM packet is -33.6324 dB.

FETCh:WLAN:AY:SUMMary:FERRor:AVERage? (Query Only)

Query the Average Frequency Error value in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:SUMMary:FERRor:AVERage?`

Returns

`<NRf>`

Units: ppm

Examples

`FETC:WLAN:AY:SUMM:FERR:AVER?` may return `-114.1332879317e-3`, which indicates that the Average Frequency Error value is -114.1332879317e-3 ppm.

FETCh:WLAN:AY:SUMMary:FERRor:AVERage:STATus? (Query Only)

Query the Average Frequency Error status in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:SUMMary:FERRor:AVERage:STATus?`

Returns

`<string>`

Examples

`FETC:WLAN:AY:SUMM:FERR:AVER:STAT?` may return `PASS`, `FAIL`, or `NA`.

FETCh:WLAN:AY:SUMMary:FERRor:HERTz? (Query Only)

Query the Frequency Error value in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax
`FETCh:WLAN:AY:SUMMary:FERRor:HERTz?`

Returns
`<NRf>`

Units: Hz or kHz

Examples
`FETC:WLAN:AY:SUMM:FERR:HERT?` may return `2.875e+3`, which indicates that the Frequency Error value is 2.875 kHz.

FETCh:WLAN:AY:SUMMary:FERRor:MAX? (Query Only)
Query the Maximum Frequency Error value in the Summary display.

Conditions
Measurement view: 802.11ay Analysis Summary

Group
Fetch commands

Syntax
`FETCh:WLAN:AY:SUMMary:FERRor:MAX?`

Returns
`<NRf>`

Units: ppm

Examples
`FETC:WLAN:AY:SUMM:FERR:MAX?` may return `-114.1332879317e-3`, which indicates that the Maximum Frequency Error is -114.1332879317e-3 ppm.

FETCh:WLAN:AY:SUMMary:FERRor:PPM? (Query Only)
Query the Frequency Error value in the Summary display.

Conditions
Measurement view: 802.11ay Analysis Summary

Group
Fetch commands

Syntax
`FETCh:WLAN:AY:SUMMary:FERRor:PPM?`

Returns
`<NRf>`

Units: ppm

Examples

FETC:WLAN:AY:SUMM:FERR:PPM? may return 575.023e-3, which indicates that the Frequency Error value is 575.023e-3 ppm.

FETCh:WLAN:AY:SUMMary:FERRor:STATus? (Query Only)

Query the Frequency Error status in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

FETCh:WLAN:AY:SUMMary:FERRor:STATus?

Returns

<string>

Examples

FETC:WLAN:AY:SUMM:FERR:STAT? may return PASS, FAIL, or NA.

FETCh:WLAN:AY:SUMMary:IQ:GAIN:IMBALance? (Query Only)

Query the IQ Gain Imbalance value in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

FETCh:WLAN:AY:SUMMary:IQ:GAIN:IMBALance?

Returns

<NRf>

Units: dB

Examples

FETC:WLAN:AY:SUMM:IQ:GAIN:IMBAL? may return 69.2700818181e-3, which indicates that the IQ Gain Imbalance value is 69.2700818181e-3 dB.

FETCh:WLAN:AY:SUMMary:IQ:ORIGin:OFFSet? (Query Only)

Query the IQ Origin Offset value in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:SUMMary:IQ:ORIGin:OFFSet?`

Returns

`<NRf>`

Units: dB

Examples

`FETC:WLAN:AY:SUMM:IQ:ORIG:OFFS?` may return `-160.0000000000`, which indicates that the IQ Origin Offset value is -160.0000000000 dB.

FETCh:WLAN:AY:SUMMary:IQ:QUADrature:ERROr? (Query Only)

Query the IQ Quadrature Error value (degrees) in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:SUMMary:IQ:QUADrature:ERROr?`

Returns

<NRf>

Units: degrees

Examples

`FETC:WLAN:AY:SUMM:IQ:QUAD:ERRO?` may return `-220.5967903137e-3`, which indicates that the IQ Quadrature Error value is -220.5967903137e-3 degrees.

FETCh:WLAN:AY:SUMMary:PACKet:AGGRegation? (Query Only)

Query the value of the Aggregation packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:SUMMary:PACKet:AGGRegation?`

Returns

`<string>`

Examples

`FETC:WLAN:AY:SUMM:PACK:AGGR?` may return `1111`, which indicates that the Aggregation value is 1111.

FETCh:WLAN:AY:SUMMary:PACKet:BEAMformed? (Query Only)

Query the value of the Beamformed packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:SUMMary:PACKet:BEAMformed?`

Returns

`<string>`

Examples

`FETC:WLAN:AY:SUMM:PACK:BEAM?` may return `0`, which indicates that the value of the Beamformed packet is 0.

FETCh:WLAN:AY:SUMMary:PACKet:BWBitmap? (Query Only)

Query the value of the Bandwidth BitMap packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:SUMMary:PACKet:BWBitmap?`

Returns

`<string>`

Examples

`FETC:WLAN:AY:SUMM:PACK:BWB?` may return `11000000`, which indicates that the value of the Bandwidth BitMap packet is 11000000.

FETCh:WLAN:AY:SUMMary:PACKet:COMPbw? (Query Only)

Query the value of the Compressed BW packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:SUMMary:PACKet:COMPbw?`

Returns

`<string>`

Examples

`FETC:WLAN:AY:SUMM:PACK:COMP?` may return `2`, which indicates that the value of the Compressed BW packet is 2.

FETCh:WLAN:AY:SUMMary:PACKet:CRC? (Query Only)

Query the value of the CRC packet information in the Summary display.

Conditions

Measurement view 802.11ay Analysis Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:SUMMary:PACKet:CRC?`

Returns

`<string>`

Examples

`FETC:WLAN:AY:SUMM:PACK:CRC?` may return `0110001001011010`, which indicates that the value of the CRC packet is 0110001001011010.

FETCh:WLAN:AY:SUMMary:PACKet:EDMGmcs? (Query Only)

Query the value of the EDMG MCS packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:SUMMary:PACKet:EDMGmcs?`

Returns

`<NRf>`

Examples

`FETC:WLAN:AY:SUMM:PACK:EDMG?` may return `5`, which indicates that the value of the EDMG MCS packet is 5.

FETCh:WLAN:AY:SUMMary:PACKet:LDPCcwlen? (Query Only)

Query the value of the additional LPDC CW Length packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:SUMMary:PACKet:LDPCcwlen?`

Returns

`<string>`

Examples

`FETC:WLAN:AY:SUMM:PACK:LDPC?` may return `0`, which indicates that the value of the LDPC CW Length packet is 0.

FETCh:WLAN:AY:SUMMary:PACKet:PRIMarychannel? (Query Only)

Query the value of the Primary Channel packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:SUMMary:PACKet:PRIMarychannel?`

Returns

`<string>`

Examples

`FETC:WLAN:AY:SUMM:PACK:PRIM?` may return `0`, which indicates that the value of the Primary Channel packet is 0.

FETCh:WLAN:AY:SUMMary:PACKet:PSDUlen? (Query Only)

Query the Length value of the packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:SUMMary:PACKet:PSDUlen?`

Returns

<string>

Examples

FETC:WLAN:AY:SUMM:PACK:PSDU? may return 1000, which indicates that the Length value of the packet is 1000.

FETCh:WLAN:AY:SUMMary:PACKet:RSSI? (Query Only)

Query the value of the last RSSI packet information in the Summary display.

The first 2 bits of the last RSSI will fetch the Guard Interval type for EDMG (00/01/10 - Short/Normal/Long).

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

FETCh:WLAN:AY:SUMMary:PACKet:RSSI?

Returns

<string>

Examples

FETC:WLAN:AY:SUMM:PACK:RSSI? may return 0000, which indicates that the last value of the RSSI packet is 0000.

FETCh:WLAN:AY:SUMMary:PACKet:SCRAmbler:INITialization? (Query Only)

Query the value of the Scrambler Initialization packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

FETCh:WLAN:AY:SUMMary:PACKet:SCRAmbler:INITialization?

Returns

<string>

Examples

FETC:WLAN:AY:SUMM:PACK:SCRA:INIT? may return 0101110, which indicates that the value of the Scrambler Initialization packet is 0101110.

FETCh:WLAN:AY:SUMMary:PACKet:SPATialstreams? (Query Only)

Query the value of the Spatial Streams packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

```
FETCh:WLAN:AY:SUMMary:PACKet:SPATialstreams?
```

Returns

```
<string>
```

Examples

`FETC:WLAN:AY:SUMM:PACK:SPAT?` may return `0`, which indicates that the value of the Spatial Streams packet is 0.

FETCh:WLAN:AY:SUMMary:PACKet:STBCapplied? (Query Only)

Query the value of the STBC Applied packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

```
FETCh:WLAN:AY:SUMMary:PACKet:STBCapplied?
```

Returns

```
<string>
```

Examples

`FETC:WLAN:AY:SUMM:PACK:STBC?` may return `0`, which indicates that the value of the STBC Applied packet is 0.

FETCh:WLAN:AY:SUMMary:PACKet:TRNLen ? (Query Only)

Query the value of the EDMG TRN Length packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

```
FETCh:WLAN:AY:SUMMary:PACKet:TRNLen?
```

Returns

<string>

Examples

FETC:WLAN:AY:SUMM:PAC:TRNL? may return 0, which indicates that the value of the EDMG TRN Length packet is 0.

FETCh:WLAN:AY:SUMMary:RCPI? (Query Only)

Query the RCPI (Received Channel Power Indicator) value in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

FETCh:WLAN:AY:SUMMary:RCPI?

Returns

<NR1>

Examples

FETC:WLAN:AY:SUMM:RCPI? may return 207.0000000000, which indicates that the RCPI value is 207.0000000000.

FETCh:WLAN:AY:SUMMary:RF:POWer? (Query Only)

Query the Radio Frequency (RF) Output Power value in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

FETCh:WLAN:AY:SUMMary:RF:POWer?

Returns

<NRf>

You must select the units in the Units tab.

Examples

FETC:WLAN:AY:SUMM:RF:POW? may return -6.3434391022, which indicates that the Radio Frequency Output Power value is -6.3434391022.

FETCh:WLAN:AY:SUMMary:SIG:QUAlity? (Query Only)

Query the Signal Quality value in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Fetch commands

Syntax

`FETCh:WLAN:AY:SUMMary:SIG:QUAlity?`

Returns

`<NRf>`

Units: rho

Examples

`FETC:WLAN:AY:SUMM:SIG:QUA?` may return `999.6323e-3`, which indicates that the Signal Quality value is 999.6323e-3 rho.

FETCh:WLAN:CONSte:MAGNitude? (Query Only)

Returns the magnitude data for the WLAN Constellation measurement.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

`FETCh:WLAN:CONSte:MAGNitude?`

Returns

Returned data is in the form

`#<num_digit><num_byte><data(1)><data(2)>..<data(n)>` where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the nth data point in dB, 4-byte little endian floating-point format specified in IEEE 488.2

Examples

`FETC:WLAN:CONS:MAGN?` might return `#510400..` (10400-byte data) for the measurement.

FETCh:WLAN:CONSte:PHASe? (Query Only)

Returns the phase values for the WLAN Constellation measurement.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

`FETCh:WLAN:CONSte:PHASe?`

Returns

Returned data is in the form

`#<num_digit><num_byte><data(1)><data(2)>..<data(n)>`

where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the nth data point in dB, 4-byte little endian floating-point format specified in IEEE 488.2

Examples

`FETCH:WLAN:CONS:PHAS?` might return `#510400..` (10400-byte data) for the measurement.

FETCh:WLAN:CONSte:TYPE? (Query Only)

Returns the type (context) for the WLAN Constellation measurement.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

`FETCh:WLAN:CONSte:TYPE?`

Returns

`#<num_digit><num_byte><data(1)><data(2)>..<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the nth data point in 16-bit little endian integers. The integers must be translated to the proper context as listed below.

Integer value Type (context):

- 1 Pilot
- 2 Data

Examples

`FETCH:WLAN:CONS:TYPE?` might return `#41352..` (1352-byte data) for the measurement. The actual data must be decoded to the context type using the table above.

FETCh:WLAN:CONSte:VALue? (Query Only)

Returns the value (decoded symbol) for the WLAN Constellation measurement.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

`FETCh:WLAN:CONSte:VALue?`

Returns

`#<num_digit><num_byte><data(1)><data(2)>..<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the nth data point in 16-bit little endian integers.

Examples

`FETCH:WLAN:CONSTE:VALUE?` might return `#41352` indicating 1352 bytes of data for the measurement.

FETCh:WLAN:CRESponse:MAGNitude? (Query Only)

Returns the magnitude data of the WLAN Channel Response measurement.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

`FETCh:WLAN:CRESponse:MAGNitude?`

Returns

Returned data is in the form

`#<num_digit><num_byte><data(1)><data(2)>..<data(n)>`

where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the nth data point in dB, 4-byte little endian floating-point format specified in IEEE 488.2

Examples

FETC:WLAN:CRES:MAGN? might return #510400.. (10400-byte data) for the measurement.

FETCh:WLAN:CRESponse:PHASe? (Query Only)

Returns the phase data of the WLAN Channel Response measurement.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

FETCh:WLAN:CRESponse:PHASe?

Returns

Returned data is in the form

#<num_digit><num_byte><data(1)><data(2)>..<data(n)>

where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point in dB, 4-byte little endian floating-point format specified in IEEE 488.2

Examples

FETCH:WLAN:CRES:PHAS? might return #510400.. (10400-byte data) representing the phase data for the measurement.

FETCh:WLAN:EVM:TRACe<x>? (Query Only)

Returns the trace data for the selected WLAN EVM trace. When <x> is 1, the parameter is Matrix.

When <x> is 2, the parameter is Average versus Symbol. When <x> is 3, the parameter is Average versus Subcarrier

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

FETCh:WLAN:EVM:TRACe<x>?

Returns

#<num_digit><num_byte><data(1)><data(2)>..<data(n)>

Where:

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

FETCH:WLAN:EVM:TRAC2? might return #510400.. (10400-byte data) for the measurement.

FETCh:WLAN:FLATness:PASS? (Query Only)

Queries whether the average power level values of the subcarriers across the signal bandwidth on the WLAN Spectral Flatness display remain within the limits defined for a particular standard.

NOTE: WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

FETCh:WLAN:FLATness:PASS?

Returns

Returns "1" in the case that all data remains within the defined limits, and "0" otherwise.

Examples

FETCH:WLAN:FLAT:PASS? might return 1, which means that the average power level values of the subcarriers across the signal bandwidth remain within the defined limits.

FETCh:WLAN:FLATness:RESult? (Query Only)

Returns the summary results of each segment of the WLAN Spectral Flatness display.

NOTE: WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

FETCh:WLAN:FLATness:RESult?

Returns

The data is returned as 12 comma delineated text values. The specific values are :

- minimum subcarrier within segment 1
- minimum subcarrier average energy deviation from the global average within segment 1
- deviation the of minimum subcarrier average from the allowed value in segment 1
- maximum subcarrier within segment 1
- maximum subcarrier average energy deviation from the global average within segment 1
- deviation of the maximum subcarrier average from the allowed value in segment 1
- minimum subcarrier within segment 2
- minimum subcarrier average energy deviation from the global average within segment 2
- deviation the of minimum subcarrier average from the allowed value in segment 2
- maximum subcarrier within segment 2
- maximum subcarrier average energy deviation from the global average within segment 2
- deviation of the maximum subcarrier average from the allowed value in segment 2

Examples

`FETCH:WLAN:FLAT:RES?` might return `-6.0000000000,-17.0626174659E-3,3.9829373825,12.0000000000,19.979260`, which represents the summary results of each segment of the WLAN Spectral Flatness display.

FETCh:WLAN:FLATness:TRACe<x>? (Query Only)

Returns the WLAN Spectral Flatness trace data. When <x> is 1, the parameter is Matrix (symbol deviation per subcarrier). When<x> is 2, the parameter is Average Deviation vs Subcarrier.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

`FETCh:WLAN:FLATness:TRACe<x>?`

Returns

`#<num_digit><num_byte><data(1)><data(2)>..<data(n)>`

Where:

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`FETCH:WLAN:FLAT:TRACE2?` might return `#510400..` (10400-byte data) for the measurement.

FETCh:WLAN:MERRor:TRACe<x>? (Query Only)

Returns the trace data for the selected WLAN Magnitude Error trace.

When <x> is 1, the parameter is Matrix.

When <x> is 2, the parameter is Average versus Symbol.

When <x> is 3, the parameter is Average versus Subcarrier

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

`FETCh:WLAN:MERRor:TRACe<x>?`

Returns

`#<num_digit><num_byte><data(1)><data(2)>..<data(n)>`

Where:

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`FETCH:WLAN:MERR:TRAC2?` might return `#510400..` (10400-byte data) for the measurement.

FETCh:WLAN:PERRor:TRACe<x>? (Query Only)

Returns the trace data for the selected WLAN Phase Error trace.

When <x> is 1, the parameter is Matrix.

When <x> is 2, the parameter is Average versus Symbol.

When <x> is 3, the parameter is Average versus Subcarrier.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

`FETCh:WLAN:PERRor:TRACe<x>?`

Returns

`#<num_digit><num_byte><data(1)><data(2)>..<data(n)>`

Where:

`<num_digit>` is the number of digits in `<num_byte>`.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

FETCH:WLAN:PERR:TRAC2? might return #510400.. (10400-byte data) for the measurement.

FETCh:WLAN:PVTime:BURSt:POWer? (Query Only)

Returns the average power of the burst packet, in dBm, for the WLAN Power vs. Time trace.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

FETCh:WLAN:PVTime:BURSt:POWer?

Returns

Floating point number that represents the average power of the burst packet, in dBm.

Examples

FETCH:WLAN:PVT:BURST:POWER? might return -1.99, which represents the value of the average power of the burst packet of -1.99 dBm.

FETCh:WLAN:PVTime:BURSt:WIDTh? (Query Only)

Returns the value of the measured time width of the burst packet from Power-On to Power-Down (or end of waveform, if that occurs before Power-Down), in seconds, for the WLAN Power vs. Time trace.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

FETCh:WLAN:PVTime:BURSt:WIDTh?

Returns

Floating point number which represents the value of the measured time width of the burst packet from Power-On to Power-Down (or end of waveform, if that occurs before Power-Down), in seconds.

Examples

FETCH:WLAN:PVT:BURST:WIDTH? might return 156.0000000000E-6, which represents a width of the burst packet of 156 µS.

FETCh:WLAN:PVTime:FALL:TIME? (Query Only)

Returns the value of the width of the falling edge, in seconds.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

`FETCh:WLAN:PVTime:FALL:TIME?`

Returns

Floating point number which represents the value of the width of the falling edge of the burst, in seconds.

Examples

`FETCH:WLAN:PVT:FALL:TIME?` might return `7e-6`, which indicates the width of the falling edge is 7 µs.

FETCh:WLAN:PVTime:FALL:TIME:STARt? (Query Only)

Returns the value of the start time of the falling edge, in seconds.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

`FETCh:WLAN:PVTime:FALL:TIME:STARt?`

Returns

Floating point number which represents the value of the start time of the falling edge of the burst, in seconds.

Examples

`FETCH:WLAN:PVT:FALL:TIME:STAR?` might return `6e-6`, which indicates that the start time of the falling edge is at 6 µs.

FETCh:WLAN:PVTime:FALL:TIME:STOP? (Query Only)

Returns the value of the stop time of the falling edge, in seconds.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

`FETCh:WLAN:PVTime:FALL:TIME:STOP?`

Returns

Floating point number which represents the value of the stop time of the falling edge of the burst, in seconds.

Examples

`FETCH:WLAN:PVT:FALL:TIME:STOP?` might return `7e-6`, which indicates the stop time of the falling edge is at 7 µs.

FETCh:WLAN:PVTime:FTIMe? (Query Only)

Returns the value of the width of the falling edge, in seconds.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

`FETCh:WLAN:PVTime:FTIMe?`

Returns

Floating point number which represents the value of the width of the falling edge of the burst, in seconds.

Examples

`FETCH:WLAN:PVT:FALL:TIME?` might return `7e-6`, which indicates the width of the falling edge is 7 µs.

FETCh:WLAN:PVTime:RISE:TIME? (Query Only)

Returns the value of the width of the rising edge, in seconds.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

`FETCh:WLAN:PVTime:RISE:TIME?`

Returns

Floating point number which represents the value of the width of the rising edge of the burst, in seconds.

Examples

`FETCH:WLAN:PVT:RISE:TIME?` might return `7e-6`, which indicates the width of the rising edge is 7 µs.

FETCh:WLAN:PVTime:RISE:TIME:STARt? (Query Only)

Returns the value of the start time of the rising edge, in seconds.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:PVTime:RISE:TIME:STARt?
```

Returns

Floating point number which represents the value of the start time of the rising edge of the burst, in seconds.

Examples

`FETCH:WLAN:PVT:RISE:TIME:STAR?` might return `6e-6`, which indicates the start time of the rising edge is at 6 µs.

FETCh:WLAN:PVTime:RISE:TIME:STOP? (Query Only)

Returns the value of the stop time of the rising edge, in seconds.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:PVTime:RISE:TIME:STOP?
```

Returns

Floating point number which represents the value of the stop time of the rising edge of the burst, in seconds.

Examples

`FETCH:WLAN:PVT:RISE:TIME:STOP?` might return `7e-6`, which indicates the stop time of the rising edge is at 7 µs.

FETCh:WLAN:PVTime:RTIMe? (Query Only)

Returns the value of the width of the rising edge, in seconds.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

`FETCh:WLAN:PVTime:RTIMe?`

Returns

Floating point number which represents the value of the width of the rising edge of the burst, in seconds.

Examples

`FETCH:WLAN:PVT:RISE:TIME?` might return `7e-6`, which indicates the width of the rising edge is 7 µs.

FETCh:WLAN:PVTime:TRACe:X? (Query Only)

Returns the horizontal values (time in seconds) for the WLAN Power vs. Time trace.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

`FETCh:WLAN:PVTime:TRACe:X?`

Returns

`#<num_digit><num_byte><data(1)><data(2)>..<data(n)>`

Where:

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`FETCH:WLAN:PVT:TRAC:X` might return `#43204` (3204-byte data), which represents the time in seconds (horizontal values) for the WLAN Power vs. Time trace.

FETCh:WLAN:PVTime:TRACe:XY? (Query Only)

Returns the horizontal value (time in seconds) and vertical value (power) for the WLAN Power vs. Time trace.

<x(n)><y(n)> is the horizontal value and vertical value pair at the nth data point.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

FETCh:WLAN:PVTime:TRACe:XY?

Returns

#<num_digit><num_byte><data(1)><data(2)>..<data(n)>

Where:

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

FETCH:WLAN:PVT:TRAC:XY? might return #43204 (3204-byte data), which represents the horizontal value (time in seconds) and vertical value (power) pair at the nth data point.

FETCh:WLAN:PVTime:TRACe[:Y]? (Query Only)

Returns the vertical values (power) for the WLAN Power vs. Time trace.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

FETCh:WLAN:PVTime:TRACe[:Y]?

Returns

#<num_digit><num_byte><data(1)><data(2)>..<data(n)>

Where:

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

FETCH:WLAN:PVT:TRAC:Y? might return #43204 (3204-byte data), which represents the vertical values (power) for the WLAN Power vs. Time trace.

FETCh:WLAN:STABle:VALUe? (Query Only)

Returns the number of bytes in the symbol table for the WLAN measurement.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:STABle:VALUe?
```

Returns

#<num_digit><num_byte>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

Examples

FETCH:WLAN:STABLE:VALUE? might return #43848 indicating that there are 3848 bytes of data in the symbol table.

FETCh:WLAN:SUMMary:BURSt:INDex? (Query Only)

Returns the index of the analyzed packet within the analysis record.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:SUMMary:BURSt:INDex?
```

Arguments

None

Returns

<NRf>

FETCh:WLAN:SUMMary:BURSt:POWer? (Query Only)

Returns the average power of all symbols in the packet, including Preamble and Data segments.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:SUMMary:BURSt:POWer?
```

Arguments

None

Returns

<NRf> data in dBm.

FETCh:WLAN:SUMMary:BURSt:POWer:CFACtor? (Query Only)

Returns the peak-to-average burst power factor in the WLAN measurement. The ratio of the highest instantaneous signal power level to the average signal power.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:SUMMary:BURSt:POWer:CFACtor?
```

Arguments

None

Returns

<NRf> data in dB.

FETCh:WLAN:SUMMary:CERRor? (Query Only)

Returns the symbol clock error in the WLAN measurement. Not available for 802.11b.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:SUMMary:CERRor?
```

Arguments

None

Returns

<NRf> data in parts per million.

FETCh:WLAN:SUMMary:CPE? (Query Only)

Returns the RMS magnitude error of the pilots over all data symbols.

Not available for 802.11b.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:SUMMary:CPE?
```

Arguments

None

Returns

<NRf> data in percent.

FETCh:WLAN:SUMMary:EVM:PEAK:ALL? (Query Only)

Returns the peak EVM value for all subcarriers in the WLAN measurement or for all chips depending on the selected standard.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:SUMMary:EVM:PEAK:ALL?
```

Arguments

None

Returns

<NRf> data in dB.

FETCh:WLAN:SUMMary:EVM:PEAK:CHIP:ALL? (Query Only)

Returns the chip at which the reported peak EVM value occurred. Available only for 802.11b

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:SUMMary:EVM:PEAK:CHIP:ALL?
```

Arguments

None

Returns

<NR1>

FETCh:WLAN:SUMMary:EVM:PEAK:CHIP:ONEKchips? (Query Only)

Returns the chip at which the reported peak EVM value over the first 1000 chips occurred.

Only available for 802.11b.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:SUMMary:EVM:PEAK:CHIP:ONEKchips?
```

Arguments

None

Returns

<NR1>

FETCh:WLAN:SUMMary:EVM:PEAK:DATA? (Query Only)

Returns the peak EVM value for the data subcarriers in the WLAN measurement. Not available for 802.11b.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:SUMMary:EVM:PEAK:DATA?
```

Arguments

None

Returns

<NRf> data in dB.

FETCh:WLAN:SUMMary:EVM:PEAK:ONEKchips? (Query Only)

Returns the peak EVM value measured over the first 1000 chips. Available only for 802.11b.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:SUMMary:EVM:PEAK:ONEKchips?
```

Arguments

None

Returns

<NRf> data in dB.

FETCh:WLAN:SUMMary:EVM:PEAK:PILots? (Query Only)

Returns the peak EVM value for the pilots in the WLAN measurement. Not available for 802.11b.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:SUMMary:EVM:PEAK:PILots?
```

Arguments

None

Returns

<NRf> data in dB.

FETCh:WLAN:SUMMary:EVM:PEAK:SCARrier:ALL? (Query Only)

Returns the subcarrier on which the ALL peak EVM value occurred.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:SUMMary:EVM:PEAK:SCARrier:ALL?
```

Arguments

None

Returns

<NR1>

FETCh:WLAN:SUMMary:EVM:PEAK:SCARrier:DATA? (Query Only)

Returns the subcarrier on which the DATA peak EVM value occurred. Not available for 802.11b.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:SUMMary:EVM:PEAK:SCARrier:DATA?
```

Arguments

None

Returns

<NR1>

FETCh:WLAN:SUMMary:EVM:PEAK:SCARrier:PILots? (Query Only)

Returns the subcarrier on which the PILOT peak EVM value occurred.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:SUMMary:EVM:PEAK:SCARrier:PILots?
```

Arguments

None

Returns

<NR1>

FETCh:WLAN:SUMMary:EVM:PEAK:SYMBol:ALL? (Query Only)

Returns the symbol on which the reported EVM value occurred.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:SUMMary:EVM:PEAK:SYMBol:ALL?
```

Arguments

None

Returns

<NR1>

FETCh:WLAN:SUMMary:EVM:PEAK:SYMBol:DATA? (Query Only)

Returns the symbol on which the DATA peak EVM value occurred. Not available for 802.11b.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:SUMMary:EVM:PEAK:SYMBol:DATA?
```

Arguments

None

Returns

<NR1>

FETCh:WLAN:SUMMary:EVM:PEAK:SYMBol:PILots? (Query Only)

Returns the symbol on which the PILOT peak EVM value occurred.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:SUMMary:EVM:PEAK:SYMBol:PILots?
```

Arguments

None

Returns

<NR1>

FETCh:WLAN:SUMMary:EVM:RMS:ALL? (Query Only)

Returns the RMS EVM data for all subcarriers in the WLAN measurement and for all chips for 802.11b.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:SUMMary:EVM:RMS:ALL?
```

Arguments

None

Returns

<NRf> RMS EVM in dB

FETCh:WLAN:SUMMary:EVM:RMS:AVERage? (Query Only)

Returns the average RMS EVM value in the WLAN measurement over the selected number of bursts.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:SUMMary:EVM:RMS:AVERage?
```

Arguments

None

Returns

<NRf> RMS EVM in dB

FETCh:WLAN:SUMMary:EVM:RMS:DATA? (Query Only)

Returns the RMS EVM data for the data subcarriers in the WLAN measurement. Not available for 802.11b.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:SUMMary:EVM:RMS:DATA?
```

Arguments

None

Returns

<NRf> RMS EVM in dB

FETCh:WLAN:SUMMary:EVM:RMS:MAXimum? (Query Only)

Returns the maximum RMS EVMvalue in the WLAN measurement over the selected number of bursts.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:SUMMary:EVM:RMS:MAXimum?
```

Arguments

None

Returns

<NRf> RMS EVM in dB

FETCh:WLAN:SUMMary:EVM:RMS:ONEKchips? (Query Only)

Returns the EVM RMSdata for the 1k chips in the WLAN measurement measured over the first 1000 chips.

Only available for 802.11b.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:SUMMary:EVM:RMS:ONEKchips?
```

Arguments

None

Returns

<NRf> RMS EVM in dB

FETCh:WLAN:SUMMary:EVM:RMS:PILots? (Query Only)

Returns the RMS EVM data for the pilot subcarriers in the WLAN measurement. Not available for 802.11b.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:SUMMary:EVM:RMS:PILots?
```

Arguments

None

Returns

<NRf> data in percent.

FETCh:WLAN:SUMMary:FERRor? (Query Only)

Returns the frequency difference between the measured carrier frequency of the signal and the measurement frequency setting.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:SUMMary:FERRor?
```

Arguments

None

Returns

<NRf> data in kHz.

FETCh:WLAN:SUMMary:HEADer? (Query Only)

Returns the received packet header information for the WLAN measurement. Applies to the 802.11b standard only.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:SUMMary:HEADer?
```

Arguments

None

Returns

Results are returned in the same left-to-right order in which they are presented in the display.

FETCh:WLAN:SUMMary:HTSig? (Query Only)

Returns the decimal values of the received packet HT-SIGNAL. Applies to the 802.11n standard only.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:SUMMary:HTSig?
```

Arguments

None

Returns

Results are returned in the same left-to-right order in which they are presented in the display.

FETCh:WLAN:SUMMary:IQOFfset? (Query Only)

Returns the average magnitude of the DC subcarrier level relative to total signal power.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:SUMMary:IQOFfset?
```

Arguments

None

Returns

<NRf> data in dB.

FETCh:WLAN:SUMMary:LSIG? (Query Only)

Returns the decimal values of the received packet (legacy) SIGNAL symbols

fields. Applies to the 802.11n standard only.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:SUMMary:LSIG?
```

Arguments

None

Returns

Results are returned in the same left-to-right order in which they are presented in the display.

FETCh:WLAN:SUMMary:PACKet? (Query Only)

Returns the type and number of symbols, EVM-RMS and average power of the Preamble and Data portions of the packet.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:SUMMary:PACKet?
```

Arguments

None

Returns

Results are returned in the same left-to-right order in which they are presented in the display.

FETCh:WLAN:SUMMary:PACKet:DMODulation? (Query Only)

Returns the modulation used in the Data symbols.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:SUMMary:PACKet:DMODulation?
```

Returns

<string> The data modulation being used.

FETCh:WLAN:SUMMary:PACKet:FORMat? (Query Only)

Returns the packet format setting for the WLAN measurement

Group

Fetch commands

Syntax

```
FETCh:WLAN:SUMMary:PACKet:FORMat?
```

Arguments

None

Returns

<string> The packet format setting.

FETCh:WLAN:SUMMary:PACKet:GUARd? (Query Only)

Returns the packet guard interval used by the DATA symbols. Not available for 802.11b.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:SUMMary:PACKet:GUARd?
```

Arguments

None

Returns

<NRf> The packet guard interval, 1/4 or 1/8.

FETCh:WLAN:SUMMary:SIG? (Query Only)

Returns the decimal values of the received packet SIGNAL symbols fields. Available only for 802.11agjp.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:SUMMary:SIG?
```

Arguments

None

Returns

Results are returned in the same left-to-right order in which they are presented in the display.

FETCh:WLAN:SUMMary:VHTSig:A? (Query Only)

Returns the decimal values of the received packet VHT-SIGNAL(A) symbols fields.

Available only for 802.11ac.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:SUMMary:VHTSig:A?
```

Arguments

None

Returns

Results are returned in the same left-to-right order in which they are presented in the display.

FETCh:WLAN:SUMMary:VHTSig:B? (Query Only)

Returns the decimal values of the received packet VHT-SIGNAL(B) symbols fields.

Available only for 802.11ac.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Fetch commands

Syntax

```
FETCh:WLAN:SUMMary:VHTSig:B?
```

Arguments

None

Returns

Results are returned in the same left-to-right order in which they are presented in the display.

*IDN? (Query Only)

Returns the analyzer identification code.

Conditions

Measurement views: All

Group

IEEE common commands

Syntax

```
*IDN?
```

Arguments

None

Returns

The analyzer identification code in the following format

`TEKTRONIX,RSAXXXXX,<serial_number>,<firmware_version>`

Where

`TEKTRONIX` indicates that the manufacturer is Tektronix.

`RSAXXXXX` is the model number (RSA5103B, RSA5106B, RSA5115B, or RSA5126B).

`<serial_number>` is the serial number.

`<firmware_version>` is the firmware version.

Examples

`*IDN?` might return the response

`TEKTRONIX,RSA5126B,B000111,FV2.6.1500`

INITiate:CONTinuous

Determines whether to place the analyzer in the single or the continuous acquisition mode.

Conditions

Measurement views: All

Group

Initiate commands

Syntax

```
INITiate:CONTinuous { OFF | ON | 0 | 1 }
INITiate:CONTinuous?
```

Related Commands

INITiate:RESume (on page 1481)

INITiate[:IMMediate] (on page 1480)

Arguments

`OFF` or 0 places the analyzer in the single acquisition mode. To initiate the acquisition, use the `INITiate[:IMMediate]` command. To stop acquisition when a trigger is not present, and to avoid a 2.7 hour time-out, send the free-run trigger command, `TRIGger:SEQuence:STATus 0`. Then, send the command `INITiate:CONTinuous OFF`. This sequence ensures a minimal delay before the `INIT:CONT OFF` command completes.

`ON` or 1 places the analyzer in the continuous acquisition mode. To initiate the acquisition, use the `INITiate[:IMMediate]` command. To stop the acquisition in the continuous mode, send the following command: `INITiate:CONTinuous OFF`

Returns

See Arguments.

Examples

`INITIATE:CONTINUOUS ON` places the analyzer in the continuous acquisition mode.

INITiate[:IMMediate] (No Query Form)

Starts input signal acquisition.

NOTE: It is an overlapped command, which does not finish executing before the next command starts executing. Use the *OPC(?) and *WAI commands to synchronize all pending operations to the execution of this command.

Conditions

Measurement views: All

Group

Initiate commands

Syntax

```
INITiate[:IMMediate]
```

Related Commands

*OPC (on page 1544)

*TRG (on page 2704)

*WAI (on page 2739)

INITiate:CONTinuous (on page 1480)

Arguments

None

Examples

`INITIATE:IMMEDIATE` starts input signal acquisition.

INITiate:RESume (No Query Form)

Restarts signal processing.

Resume is only applicable when FastFrame is enabled, and when the instrument is stopped while analyzing the frames. When the instrument is stopped, measurements will finish processing and the system might stop before all of the frames of an acquisition are processed (when acquisition is stopped, the system stops as soon as all measurements say they are finished processing the current acquisition). However, the instrument saves all of the frames to the Acquisition History (Data Cache), and when INITiate:RESume is sent, it will begin processing at the first frame of the last acquisition that was not processed (the one after the last frame processed), and process all of these remaining frames before starting a new acquisition. If INITiate is sent instead, any remaining frames are discarded and a new acquisition is started immediately. If FastFrame is off, or there are no remainder frames, or the acquisition was aborted rather than stopped, then INITiate:RESume and INITiate are the same.

Conditions

Measurement views: All

FastFrame enabled

Group

Initiate commands

Syntax

```
INITiate:RESume
```

Related Commands

*OPC (on page 1544)

*TRG (on page 2704)

*WAI (on page 2739)

INITiate:CONTinuous (on page 1480)

INITiate[:IMMediate] (on page 1480)

Arguments

None

Examples

INITIATE:RESUME (on page 1481) restarts input signal acquisition.

Input commands

INPut:CORRection:EXTernal:EDIT<x>:INTerpolation

Sets or queries the interpolation setting to use with the indicated external gain table.

Conditions

Measurement views: All

Group

Input commands

Syntax

```
INPut:CORRection:EXTernal:EDIT<x>:INTerpolation { LINear | LOGarithmic }
INPut:CORRection:EXTernal:EDIT<x>:INTerpolation
```

Related Commands

INPut:CORRection:EXTernal:EDIT<x>:NEW (on page 1482)

Arguments

`LINear` select when the frequency scale of the spectrum or spurious measurements are linear.

`LOGarithmic` select when the frequency scale of the spectrum is logarithmic.

Examples

`INPut:CORRection:EXTernal:EDIT1:INTerpolation LINear` uses linear interpolation when applying External correction Table one (1) to the chosen waveforms.

INPut:CORRection:EXTernal:EDIT<x>:LABel

Sets or queries the name of the external loss table.

The parameter <x> = 1 to 3 represent the External Loss Table 1 to 3, respectively.

NOTE: This command has replaced `CALIbration:CORRection:EXTernal:EDIT<x>:LABEL`

Conditions

Measurement views: All

Group

Input commands

Syntax

```
INPut:CORRection:EXTernal:EDIT<x>:LABel <***Set NAME variable****>
INPut:CORRection:EXTernal:EDIT<x>:LABel?
```

Arguments

<***Set NAME variable****>::=<string> specifies the name of the external loss table.

Examples

INPut:CORRection:EXTernal:EDIT<x>:LABel 'Sample Table 1' names the External Loss Table 1 "Sample Table 1".

INPut:CORRection:EXTernal:EDIT<x>:NEW

Creates the indicated external loss table from a series of frequency and gain value pairs.

NOTE: This command has replaced CALIbration:CORRection:EXTernal:EDIT<x>:NEW

Conditions

Measurement views: All

Group

Input commands

Syntax

```
INPut:CORRection:EXTernal:EDIT<x>:NEW
   <freq(1)>,<gain(1)>,<freq(2)>,<gain(2)>,...,<freq(n)>,<gain(n)>
INPut:CORRection:EXTernal:EDIT<x>:NEW?
```

Related Commands

INPut:CORRection:EXTernal:TYPE (on page 1485)

Arguments

<freq(n)>,<gain(n)> specifies a pair of frequency (<NR3> in Hz) and gain (+ | -<NR3> in dB) in the external correction table.

NOTE: Do not include a unit suffix (such as Hz, dBm, or dB) with the arguments for this command.

The setting range is:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Loss: -50 to +30 dB. Negative values indicate loss and positive values indicate gain. Resolution is 0.1 dB.

Examples

INPut:CORRection:EXTernal:EDIT?:NEW 1.0E+9,-2.2,-1.5E+9,–2.3 creates the External correction Table specifying a loss of 2.2 dB at 1 GHz and a loss of 2.3 dB at 1.5 GHz.

INPut:CORRection:EXTernal:EDIT<x>:STATe

Determines whether to enable or disable the external loss table.

The parameter <x> = 1 to 3 represent the External Loss Table 1 to 3, respectively.

NOTE: This command has replaced `CALIbration:CORRection:EXTernal:EDIT<x>:STATe`

Conditions

Measurement views: All

Group

Input commands

Syntax

```
INPut:CORRection:EXTernal:EDIT<x>:STATe { OFF | ON | 0 | 1 }
INPut:CORRection:EXTernal:EDIT<x>:STATe?
```

Arguments

`OFF` or `0` disables the external loss table.

`ON` or `1` enables the external loss table.

NOTE: You can enable one or more tables at the same time.

Examples

`INPut:CORRection:EXTernal:EDIT3:STATe?` ON enables the External Loss Table 3.

INPut:CORRection:EXTernal:GAIN[:MAGNitude]

Sets or queries the external gain value. It can be enabled or disabled using the `INPut:CORRection:EXTernal:GAIN:STATe` command.

NOTE: This command has replaced `CALIbration:CORRection:EXTernal:GAIN[:MAGNitude]`

Conditions

Measurement views: All

Group

Input commands

Syntax

```
INPut:CORRection:EXTernal:GAIN[:MAGNitude] <value>
INPut:CORRection:EXTernal:GAIN[:MAGNitude]?
```

Arguments

`<value>::=<NRf>` specifies the external gain value. Range: -50 to +30 dB.

Examples

`INPut:CORRection:EXTernal:GAIN[:MAGNitude] <value>` -10 sets the external gain to -10 dB.

INPut:CORRection:EXTernal:GAIN:STATe

Determines whether to enable or disable the external gain value.

NOTE: This command has replaced `CALIbration:CORRection:EXTernal:GAIN:STATe`.

Conditions

Measurement views: All

Group

Input commands

Syntax

```
INPut:CORRection:EXTernal:GAIN:STATe { OFF | ON | 0 | 1 }
INPut:CORRection:EXTernal:GAIN:STATe?
```

Related Commands

INPut:CORRection:EXTernal:GAIN[:MAGNitude] (on page 1484)

Arguments

OFF or 0 disables the external gain value.

ON or 1 enables the external gain value.

Examples

`INPut:CORRection:EXTernal:GAIN:STATe ON` enables the external gain value.

INPut:CORRection:EXTernal:TYPE

Sets or queries the data type to use when applying the external loss table corrections.

NOTE: This command has replaced `CALIbration:CORRection:EXTernal:TYPE`.

Conditions

Measurement views: All

Group

Input commands

Syntax

```
INPut:CORRection:EXTernal:TYPE { TRACe | DATA }
INPut:CORRection:EXTernal:TYPE
```

Related Commands

INPut:CORRection:EXTernal:EDIT<x>:NEW (on page 1483)

Arguments

`TRACe` selects traces in the Spectrum, Spectrogram, Spurious, and Amplitude versus Time views. Selecting `TRACe` disables the `INPut:CORRection:EXTernal:GAIN:STATe` command.

`DATA` selects all acquired data. Selecting `DATA` enables the `INPut:CORRection:EXTernal:GAIN:STATe` command.

Examples

`INPUT:CORRECTION:EXTERNAL:TYPE TRACe` selects traces to apply the external loss table corrections.

INPut:{MLEVel|RLEVel}

Sets or queries the reference level.

Conditions

Measurement views: All

Group

Input commands

Syntax

```
INPut:{MLEVel|RLEVel} <value>
INPut:{MLEVel|RLEVel}?
```

Arguments

`<value>::<NRf>` specifies the reference level. Range: -170 to +50 dBm.

Examples

`INPUT:RLEVEL` -10 sets the reference level to -10 dBm.

INPut[:RF]:ATTenuation

Sets or queries the input attenuation. Programming a specified attenuation sets `INPut[:RF]:ATTenuation:AUTO` OFF.

Conditions

Measurement views: All

Group

Input commands

Syntax

```
INPut[:RF]:ATTenuation <value>
INPut[:RF]:ATTenuation?
```

Related Commands

INPut[:RF]:ATTenuation:AUTO (on page 1486)

Arguments

`<value>::<NR1>` specifies the input attenuation.

Range: 0 to 75 dB in 5 dB steps.

Examples

`INPUT:RF:ATTENUATION 20` sets the input attenuation to 20 dB.

INPut[:RF]:ATTenuation:AUTO

Determines whether to set the input attenuation automatically or manually.

Conditions

Measurement views: All

Group

Input commands

Syntax

```
INPut[:RF]:ATTenuation:AUTO { OFF | ON | 0 | 1 }
INPut[:RF]:ATTenuation:AUTO?
```

Arguments

`OFF` or 0 specifies that the input attenuation is set manually. To set it, use the `INPut[:RF]:ATTenuation` command.

`ON` or 1 specifies that the input attenuation is set automatically according to the reference level.

Examples

`INPUT:RF:ATTENUATION:AUTO ON` specifies that the input attenuation is set automatically.

INPut[:RF]:ATTenuation:MONitor:STATe

Determines whether to enable or disable to monitor attenuator use.

Conditions

Measurement views: All

Group

Input commands

Syntax

```
INPut[:RF]:ATTenuation:MONitor:STATe { OFF | ON | 0 | 1 }
INPut[:RF]:ATTenuation:MONitor:STATe?
```

Arguments

`OFF` or `0` disables to monitor attenuator use.

`ON` or `1` enables to monitor attenuator use.

Examples

`INPUT:RF:ATTENUATION:MONITOR:STATE ON` enables to monitor attenuator use.

INPut[:RF]:GAIN:STATe

Determines whether to enable or disable the internal preamp.

Conditions

Measurement views: All (Option 50 or 51 only)

Group

Input commands

Syntax

```
INPut[:RF]:GAIN:STATe { OFF | ON | 0 | 1 }
INPut[:RF]:GAIN:STATe?
```

Arguments

`OFF` or `0` disables the internal preamp.

`ON` or `1` enables the internal preamp.

Examples

`INPUT:RF:GAIN:STATE ON` enables the internal preamp.

INPut:TYPE

Sets or queries the source type.

Condition

Measurement views: All

Group

Input commands

Syntax

```
INPut:TYPE <value> { RF | IQ | DIFFiq}
INPut:TYPE?
```

Arguments

`<value>::<NRf>` specifies the source type.

Examples

`INPUT:TYPE RF` will set the source type to RF.

Memory commands

MMEMory:{AM|FM|PM}:LOAD:TRACe (No Query Form)

Loads the AM/FM/PM measurement trace data from the specified file.

Conditions

Measurement views: AM, FM, PM

Group

Mass Memory commands

Syntax

```
MMEMory:{AM|FM|PM}:LOAD:TRACe <file_name>
```

Arguments

`<file_name>::<string>` specifies the file from which to load trace data.The file extension is .AmVsTime (AM), .FmVsTime (FM), or .PmVsTime (PM). You can omit the extension.

Examples

`MMEMORY:AM:LOAD:TRACE "Sample1"` loads the trace data from the *Sample1.AmVsTime* file in the AM measurement.

MMEMory:{AM|FM|PM}:SHOW:TRACe<x>

Enables display of a recalled trace file in Trace<x>. The result is the same as selecting “Show recalled trace” in the Recall traces dialog box or selecting Show in the Trace Overview display.

Conditions

Measurement views: AM, FM, PM

Group

Mass Memory commands

Syntax

```
MMEMory:{AM|FM|PM}:SHOW:TRACe<x> { OFF | ON | 0 | 1 }
MMEMory:{AM|FM|PM}:SHOW:TRACe<x>?
```

Related Commands

MMEMory:{AM|FM|PM}:LOAD:TRACe (on page 1488)

Arguments

`OFF` or 0 disables display of the recalled trace.

`ON` or 1 enables display of the recalled trace.

Examples

`MMEMORY:FM:SHOW:TRACE2 ON` enables display of the recalled data loaded in Trace2.

MMEMory:{AM|FM|PM}:STORe:TRACe (No Query Form)

Stores the AM/FM/PM measurement trace data in the specified file.

Conditions

Measurement views: AM, FM, PM

Group

Mass Memory commands

Syntax

```
MMEMory:{AM|FM|PM}:STORe:TRACe <file_name>
```

Arguments

`<file_name>::<string>` specifies the file from which to load trace data.The file extension is .AmVsTime (AM), .FmVsTime (FM), or .PmVsTime (PM). You can omit the extension.

Examples

`MMEMORY:AM:STORE:TRACE "Sample1"` stores the trace data in the *Sample1.AmVsTime* file in the AM measurement.

MMEMory:AUDio:LOAD:FILTer (No Query Form)

Loads a user-defined audio filter from the specified file.

Conditions

Measurement views: Audio Spectrum

Group

Mass Memory commands

Syntax

`MMEMory:AUDio:LOAD:FILTer <file_name>`

Related Commands

[SENSe]:AUDio:FILTer:SOURce? (on page 1971)

Arguments

`<file_name> ::= <string>` specifies the filter file to load.

Examples

`MMEMORY:AUDIO:LOAD:FILTER "filter4"` loads the user-defined filter4 file for the measurement.

MMEMory:AVTime[:MEASview<y>]:LOAD:TRACe<x> (No Query Form)

Loads the Amplitude versus Time trace data from the specified file. The parameter `<x>` = 1 to 3; Trace 4 (math trace) is invalid.

Conditions

Measurement views: Amplitude versus Time

Group

Mass Memory commands

Syntax

`MMEMory:AVTime[:MEASview<y>]:LOAD:TRACe<x> <file_name>`

Arguments

`<file_name>::<string>` specifies the file from which to load trace data.The file extension is .AmplVsTime. You can omit the extension.

The parameter `<x>` = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

The parameter `<y>`=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`MMEMORY:AVTIME:MEASVIEW1:LOAD:TRACE1 "Sample1"` loads Trace 1 data on Amplitude versus Time Measurement View 1 from the *Sample1.AmplVsTime* file.

MMEMory:AVTime[:MEASview<y>]:SHOW:TRACe<x>

Enables display of a recalled trace file in Trace<x>. The result is the same as selecting “Show recalled trace” in the Recall traces dialog box or selecting Show in the Trace Overview display.

Conditions

Measurement views: Amplitude versus Time

Group

Mass Memory commands

Syntax

```
MMEMory:AVTime[:MEASview<y>]:SHOW:TRACe<x> { OFF | ON | 0 | 1 }
MMEMory:AVTime[:MEASview<y>]:SHOW:TRACe<x>?
```

Related Commands

MMEMory:AVTime[:MEASview<y>]:LOAD:TRACe<x> (on page 1490)

Arguments

`OFF` or `0` disables display of the recalled trace.

`ON` or `1` enables display of the recalled trace.

The parameter `<x>` = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter `<y>`=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`MMEMORY:AVTIME:MEASVIEW1:SHOW:TRACE2 ON` enables display of the recalled data loaded in Trace2 from Amplitude versus Time Measurement View 1.

MMEMory:AVTime[:MEASview<y>]:STORe:PLOT

Stores the Amplitude versus Time trace data in the specified file. The parameter <x> = 1 to 4; All traces are valid.

Conditions

Measurement views: Amplitude versus Time

Group

Mass Memory commands

Syntax

```
MMEMory:AVTime[:MEASview<y>]:STORe:PLOT<file_name>
```

Arguments

`<file_name>::<string>` specifies the file in which to store trace data.The file extension is .AmplVsTime. You can omit the extension.

The parameter `<y>`=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`MMEMORY:AVTIME:MEASVIEW1:STORE:PLOT "Sample1"` stores Trace 1 data from Amplitude versus Time Measurement View 1 in the *Sample1.AmplVsTime* file.

MMEMory:AVTime[:MEASview<y>]:STORe:PLOT:DONe

Stores the Amplitude versus Time trace data in the specified file. The parameter <x> = 1 to 4; All traces are valid.

Conditions

Measurement views: Amplitude versus Time

Group

Mass Memory commands

Syntax

`MMEMory:AVTime[:MEASview<y>]:STORe:PLOT:DONe <file_name>`

Arguments

`<file_name>::<string>` specifies the file in which to store trace data.The file extension is .AmplVsTime. You can omit the extension.

The parameter `<y>`=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`MMEMORY:AVTIME:MEASVIEW1:STORE:PLOT:DONE "Sample1"` stores Trace 1 data from Amplitude versus Time Measurement View 1 in the *Sample1.AmplVsTime* file.

MMEMory:AVTime[:MEASview<y>]:STORe:TRACe<x> (No Query Form)

Stores the Amplitude versus Time trace data in the specified file. The parameter <x> = 1 to 4; All traces are valid.

Conditions

Measurement views: Amplitude versus Time

Group

Mass Memory commands

Syntax

`MMEMory:AVTime[:MEASview<y>]:STORe:TRACe<x> <file_name>`

Arguments

`<file_name>::<string>` specifies the file in which to store trace data.The file extension is .AmplVsTime. You can omit the extension.

The parameter `<x>` = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter `<y>`=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`MMEMORY:AVTIME:MEASVIEW1:STORE:TRACE1 "Sample1"` stores Trace 1 data from Amplitude versus Time Measurement View 1 in the *Sample1.AmplVsTime* file.

MMEMory:CALibration:LOAD:CORRection:EXTernal:EDIT<x> (No Query Form)

Loads the external loss table from a specified file.

The parameter `<x>` = 1 to 3 represent External Loss Table 1 to 3, respectively.

Conditions

Measurement views: All

Group

Mass Memory commands

Syntax

`MMEMory:CALibration:LOAD:CORRection:EXTernal:EDIT<x> <file_name>`

Arguments

`<file_name>::<string>` specifies the file to load the external loss table from.The file extension is .csv. You can omit the extension.

Examples

`MMEMORY:CALIBRATION:LOAD:CORRECTION:EXTERNAL:EDIT1 "Table1"` loads the External Loss Table 1 from the *Table1.csv* file.

MMEMory:CALibration:STORe:CORRection:EXTernal:EDIT<x> (No Query Form)

Stores the external loss table to a specified file.

The parameter `<x>` = 1 to 3 represent External Loss Table 1 to 3, respectively.

Conditions

Measurement views: All

Group

Mass Memory commands

Syntax

`MMEMory:CALibration:STORe:CORRection:EXTernal:EDIT<x> <file_name>`

Arguments

`<file_name>::<string>` specifies the file to store the external loss table to.The file extension is .csv. You can omit the extension.

Examples

`MMEMORY:CALIBRATION:STORE:CORRECTION:EXTERNAL:EDIT1 "Table1"` stores the External Loss Table 1 to the *Table1.csv* file.

MMEMory:CCDF:LOAD:TRACe<x> (No Query Form)

Loads the CCDF trace data from the specified file.

Conditions

Measurement views: CCDF

Group

Mass Memory commands

Syntax

`MMEMory:CCDF:LOAD:TRACe<x> <file_name>`

Arguments

`<file_name>::<string>` specifies the file from which to load trace data.The file extension is .CCDF. You can omit the extension.

Examples

`MMEMORY:CCDF:LOAD:TRACE1 "Sample1"` loads Trace 1 data from the *Sample1.CCDF* file.

MMEMory:CCDF:SHOW:TRACe<x>

Enables display of a recalled trace file in Trace<x>. The result is the same as selecting “Show recalled trace” in the Recall traces dialog box or selecting Show in the Trace Overview display.

Conditions

Measurement views: CCDF

Group

Mass Memory commands

Syntax

`MMEMory:CCDF:SHOW:TRACe<x> { OFF | ON | 0 | 1 } MMEMory:CCDF:SHOW:TRACe<x>?`

Related Commands

MMEMory:CCDF:LOAD:TRACe<x> (on page 1493)

Arguments

`OFF` or 0 disables display of the recalled trace.

`ON` or 1 enables display of the recalled trace.

Examples

`MMEMORY:CCDF:SHOW:TRACE2 ON` enables display of the recalled data loaded in Trace2.

MMEMory:CCDF:STORe:TRACe<x> (No Query Form)

Stores the CCDF trace data in the specified file.

Conditions

Measurement views: CCDF

Group

Mass Memory commands

Syntax

`MMEMory:CCDF:STORe:TRACe<x> <file_name>`

Arguments

`<file_name>::<string>` specifies the file in which to store trace data.The file extension is .CCDF. You can omit the extension.

Examples

`MMEMORY:CCDF:STORE:TRACE1 "Sample1"` stores Trace 1 data in the *Sample1.CCDF* file.

MMEMory:DDEMod[:SOURce<x>]:LOAD:FILTer:MEASurement:UOTHer (No query form)

Loads a user-defined measurement filter from the specified file of the measurement source.

Group

Mass Memory commands

Syntax

`MMEMory:DDEMod[:SOURce<x>]:LOAD:FILTer:MEASurement:UOTHer <file_name>`

Related Commands

[SENSe]:DDEMod[:SOURce<x>]:FILTer:REFerence (on page 2033)

[SENSe]:DDEMod[:SOURce<x>]:FILTer:MEASurement (on page 2032)

Arguments

`<x>` = 1 to 8, representing Range of Source.

`<file_name>::<string>` specifies the filter file to load.

Examples

`MMEMORY:DDEMOD:SOURCE1:LOAD:FILTER:MEASUREMENT:UOTHER "Table2"` loads User other filter from the file Table2.csv of the measurement source 1.

MMEMory:DDEMod[:SOURce<x>]:LOAD:FILTer:MEASurement:USER<y> (No query form)

Loads the user-defined measurement filter from the specified file of the measurement source. The parameter <y> specifies one of the User Meas Filters 1 to 3. The filter file must have a .csv file extension, though you may omit the file extension in the command line.

Group

Mass Memory commands

Syntax

`MMEMory:DDEMod[:SOURce<x>]:LOAD:FILTer:MEASurement:USER<y> <file_name>`

Related Commands

[SENSe]:DDEMod[:SOURce<x>]:FILTer:REFerence (on page 2033)

[SENSe]:DDEMod[:SOURce<x>]:FILTer:MEASurement (on page 2032)

Arguments

`<x>` = 1 to 8, representing Range of Source.

`<file_name>::<string>` specifies the filter file to load.

Examples

`MMEMORY:DDEMOD:SOURCE1:LOAD:FILTER:MEASUREMENT:USER1 "Table1"` loads User Meas Filter 1 from the file Table1.csv of the measurement source 1.

MMEMory:DDEMod[:SOURce<x>]:LOAD:FILTer:REFerence:UOTHer (No query form)

Loads the specified user-defined reference filter, UOTHer, from a specified file of the measurement source.

Group

Mass Memory commands

Syntax

`MMEMory:DDEMod[:SOURce<x>]:LOAD:FILTer:REFerence:UOTHer <file_name>`

Related Commands

[SENSe]:DDEMod[:SOURce<x>]:FILTer:REFerence (on page 2033)

[SENSe]:DDEMod[:SOURce<x>]:FILTer:MEASurement (on page 2032)

Arguments

`<x>` = 1 to 8, representing Range of Source.

`<file_name>::<string>` specifies the reference filter file to load. The file extension is type .csv.

Examples

`MMEMORY:DDEMOD:SOURCE1:LOAD:FILTER:REFERENCE:UOTHER "FinalFilt"` loads the filter file `FinalFilt.csv` into the filter reference UOTHER of the measurement source 1.

MMEMory:DDEMod[:SOURce<x>]:LOAD:FILTer:REFerence:USER<y> (No query form)

Loads the specified user-defined reference filter, USER1 | 2 | 3 from a specified file of the measurement source.

Group

Mass Memory commands

Syntax

`MMEMory:DDEMod[:SOURce<x>]:LOAD:FILTer:REFerence:USER<y> <file_name>`

Related Commands

[SENSe]:DDEMod[:SOURce<x>]:FILTer:REFerence (on page 2033)

Arguments

`<x>` = 1 to 8, representing Range of Source.

`<file_name>::<string>` specifies the reference filter file to load. The file extension is type .csv.

Examples

`MMEMORY:DDEMOD:SOURCE1:LOAD:FILTER:REFERENCE:USER2 "DUTFilt2"` loads the filter file *DUTfilt2.csv* into the filter reference USER2 of the measurement source 1.

MMEMory:DDEMod[:SOURce<x>]:LOAD:SYMBol:MAP (No query form)

Loads the specified symbol map filename of the measurement source. The setting correlates to the field, User Symbol Map (per Modulation Type) on the Advanced Params tab in the Digital Modulation settings.

Group

Mass Memory commands

Syntax

`MMEMory:DDEMod[:SOURce<x>]:LOAD:SYMBol:MAP <file_name>`

Related Commands

[SENSe]:DDEMod[:SOURce<x>]:MODulation:TYPE (on page 2037)

Arguments

`<x>` = 1 to 8, representing Range of Source.

`<file_name>::<string>` specifies the symbol map file to load. The file extension is type .txt.

Examples

`MMEMORY:DDEMOD:SOURCE1:LOAD:SYMBOL:MAP "C:\usymmap.txt"` identifies the file to load for the symbol map of the measurement source 1.

MMEMory:DPX:LOAD:TRACe<x> (No Query Form)

Loads the DPX spectrum trace data from the specified file.

The parameter <x> = 1, 2, 3, or 5; TRACe4 (math trace) is invalid.

Conditions

Measurement views: DPX spectrum

Group

Mass Memory commands

Syntax

`MMEMory:DPX:LOAD:TRACe<x> <file_name>`

Arguments

`<file_name>::<string>` specifies the file from which to load trace data. The file extension is .dpt. You can omit the extension.

Examples

`MMEMORY:DPX:LOAD:TRACE1 "Sample1"` loads Trace 1 data from the *Sample1.dpt* file.

MMEMory:DPX:SHOW:TRACe<x>

Enables display of a recalled trace file in Trace<x>. The result is the same as selecting “Show recalled trace” in the Recall traces dialog box or selecting Show in the Trace Overview display.

Conditions

Measurement views: DPX spectrum

Group

Mass Memory commands

Syntax

```
MMEMory:DPX:SHOW:TRACe<x> { OFF | ON | 0 | 1 }
MMEMory:DPX:SHOW:TRACe<x>?
```

Related Commands

MMEMory:DPX:LOAD:TRACe<x> (on page 1497)

Arguments

`OFF` or 0 disables display of the recalled trace.

`ON` or 1 enables display of the recalled trace.

Examples

`MMEMORY:DPX:SHOW:TRACE2 ON` enables display of the recalled data loaded in Trace2.

MMEMory:DPX:STORe:TRACe<x>

Stores the DPX spectrum trace data in the specified file. The parameter <x> = 1 to 5; All traces are valid.

Conditions

Measurement views: DPX spectrum

Group

Mass Memory commands

Syntax

```
MMEMory:DPX:STORe:TRACe<x> <file_name>
```

Arguments

`<file_name>::<string>` specifies the file in which to store trace data.The file extension is .dpt. You can omit the extension.

Examples

`MMEMORY:DPX:STORE:TRACE1 "Sample1"` stores Trace 1 data in the *Sample1.dpt* file.

MMEMory:DPX:TDM:FREQuency:TXBer:LOAD:PATTern (No Query Form)

Loads the Tx BER User data pattern from the specified file.

Conditions

Measurement views: DPX:Frequency

Group

Mass Memory commands

Syntax

```
MMEMory:DPX:TDM:FREQuency:TXBer:LOAD:PATTern <file_name>
```

Related Commands

[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern (on page 2070)

[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern:REPeat (on page 2071)

Arguments

`<file_name>::<string>` that specifies the file (path) from which to load the User data pattern. The file type containing the data pattern must be in CSV format. See the instrument Help for details on creating a user pattern file.

Examples

`MMEMORY:DPX:TDM:FREQUENCY:TXBER:LOAD:PATTERN "C:\RSA5100B Files\Example Files\SampleUserPattern.csv"` loads the user pattern defined in *SampleUserPattern.csv*.

MMEMory:EMC:ACCEssory<x>:LOAD:TABLe (No Query Form)

Loads the loss table of the specified accessory from the specified file in the EMC view.

`<x>` = <NR1> 1 to 6

Conditions

Measurement view: EMC - EMI Display

Group

Mass Memory commands

Syntax

```
MMEMory:EMC:ACCEssory<x>:LOAD:TABLe <file name>
```

Arguments

`<file name> = <String>` specifies the file to load the loss table with extension as .emcemi

Returns

None

Examples

`MMEMory:EMC:ACCEssory1:LOAD:TABLe "C:\SignalVu-PC Files\table.csv"` loads the loss table for accessory 1 from the table *.csv* file.

MMEMory:EMC:ACCEssory<x>:STORe:TABLe (No Query Form)

Stores the loss table of the specified accessory into the specified file in the EMC view.

`<x>` = <NR1> 1 to 6

Conditions

Measurement view: EMC - EMI Display

Group

Mass Memory commands

Syntax

```
MMEMory:EMC:ACCEssory<x>:STORe:TABLe <file name>
```

Arguments

`<file name> = <String>` specifies the file to store the loss table with extension as .csv

Returns

None

Examples

`MMEMory:EMC:ACCEssory2:STORe:TABLe "C:\SignalVu-PC Files\sample.csv"` stores the loss table for the accessory 2 in the sample.csv file.

MMEMory:EMC:AMBIent:TRACe:FILE (No Query Form)

Recalls the ambient trace from the file in EMC view.

Conditions

Measurement view: EMC - EMI Display

Group

Mass Memory commands

Syntax

```
MMEMory:EMC:AMBIent:TRACe:FILE <file name>
```

Arguments

`<file name> = <String>` specifies the file with extension as .emcemi to recall the ambient trace.

Returns

None

Examples

`MMEMORY:EMC:AMBIENT:TRACE:FILE "C:\SignalVu-PC Files\ambient.emcemi"` recalls the ambient trace from the ambient.emcemi file.

MMEMory:EMC:AMBIent:TRACe:SHOW (No Query Form)

Sets or queries to enable or disable show recalled trace checkbox of ambient trace in EMC view.

Conditions

Measurement view: EMC - EMI Display

Group

Mass Memory commands

Syntax

```
MMEMory:EMC:AMBIent:TRACe:SHOW {0 | OFF | 1 | ON}
MMEMory:EMC:AMBIent:TRACe:SHOW?
```

Arguments

`0` or OFF disable the show recalled trace check box of ambient trace

`1` or ON enable the show recalled trace check box of ambient trace

Returns

`<Boolean>` 0,1

Examples

`MMEMORY:EMC:AMBIENT:TRACE:SHOW?` might return `1`, indicating that the show recalled trace check box of ambient trace is enabled.

MMEMory:EMC:AMBIent:TRACe:STORe (No Query Form)

Stores the ambient trace into the specified file in the EMC view.

Conditions

Measurement view: EMC - EMI Display

Group

Mass Memory commands

Syntax

`MMEMory:EMC:AMBIent:TRACe:STORe <file name>`

Arguments

`<file name> = <String>` specifies the file to store the ambient trace with extension as .emcemi

Returns

None

Examples

`MMEMORY:EMC:AMBIENT:TRACE:STORE "C:\SignalVu-PC Files\ambient.emcemi"` stores the ambient trace in the ambient.emcemi file.

MMEMory:EMC:MATH:TRACe:FILE (No Query Form)

Recalls the math trace from the file in EMC view.

Conditions

Measurement view: EMC - EMI Display

Group

Mass Memory commands

Syntax

`MMEMory:EMC:MATH:TRACe:FILE <file name>`

Arguments

`<file name> = <String>` specifies the file with extension as .emcemi to recall the math trace.

Returns

None

Examples

`MMEMORY:EMC:MATH:TRACE:FILE "C:\SignalVu-PC Files\math.emcemi"` recalls the math trace from the math.emcemi file.

MMEMory:EMC:MATH:TRACe:SHOW

Sets or queries to enable and disable the show recalled trace check box of math trace in EMC view.

Conditions

Measurement view: EMC - EMI Display

Group

Mass Memory commands

Syntax

```
MMEMory:EMC:MATH:TRACe:SHOW {0 | OFF | 1 | ON}
MMEMory:EMC:MATH:TRACe:SHOW?
```

Arguments

`0` or OFF disable the show recalled trace check box of math trace

`1` or ON enable the show recalled trace check box of math trace

Returns

`<Boolean>` 0,1

Examples

`MMEMORY:EMC:MATH:TRACE:SHOW` might return `0`, indicating that show recalled trace check box of math trace is disabled.

MMEMory:EMC:MATH:TRACe:STORe (No Query Form)

Stores the math trace into the specified file in the EMC view.

Conditions

Measurement view: EMC - EMI Display

Group

Mass Memory commands

```
Syntax
MMEMory:EMC:MATH:TRACe:STORe <file name>
```

Arguments

`<file name> = <String>` specifies the file to store the math trace with extension as .emcemi

Returns

None

Examples

`MMEMORY:EMC:MATH:TRACE:STORE "C:\SignalVu-PC Files\math.emcemi"` stores the math trace in the math.emcemi file.

MMEMory:EMC:RANGe:LOAD:TABLe (No Query Form)

Loads the Ranges table from the specified file in the EMC view.

Conditions

Measurement view: EMC - EMI Display

Group

Mass Memory commands

Syntax

`MMEMory:EMC:RANGe:LOAD:TABLe <file name>`

Arguments

`<file name> = <String>` specifies the file to load the range table with extension as .csv

Returns

None

Examples

`MMEMORY:EMC:RANGE:LOAD:TABLE "C:\SignalVu-PC Files\table.csv"` loads the ranges table from the table.csv file.

MMEMory:EMC:RANGe:STORe:TABLe (No Query Form)

Stores the Ranges table into the specified file in the EMC view.

Conditions

Measurement view: EMC - EMI Display

Group

Mass Memory commands

Syntax

`MMEMory:EMC:RANGe:STORe:TABLe <file name>`

Arguments

`<file name> = <String>` specifies the file to store the range table with extension as .csv

Returns

None

Examples

`MMEMORY:EMC:RANGE:STORE:TABLE "C:\SignalVu-PC Files\sample.csv"` stores the ranges table in the sample.csv file.

MMEMory:EMC:RECAll:TRACe<x>:FILE (No Query Form)

Recalls the specified trace from the file in EMC view.

<x> = <NR1> Either 1,2,3,4,5 (trace number)

Conditions

Measurement view: EMC - EMI Display

Group

Mass Memory commands

```
Syntax
MMEMory:EMC:RECAll:TRACe<x>:FILE <file name>
```

Arguments

`<file name> = <String>` specifies the file with extension as .emcemi to recall the trace.

Returns

None

Examples

`MMEMory:EMC:RECAll:TRACe1:FILE "C:\SignalVu-PC Files\sampletrace.emcemi"` recalls the trace 1 from the sampletrace.emcemi file.

MMEMory:EMC:RECAll:TRACe<x>:SHOW (No Query Form)

Sets or queries to show or hide the specified recalled trace in EMC view.

<x> = <NR1> Either 1,2,3,4,5 (trace number)

Conditions

Measurement view: EMC - EMI Display

Group

Mass Memory commands

Syntax

```
MMEMory:EMC:RECAll:TRACe<x>:SHOW {0 | OFF | 1 | ON}
MMEMory:EMC:RECAll:TRACe<x>:SHOW?
```

Arguments

`0` or OFF disables the show check box and hides the recalled trace

`1` or ON enables the show check box and displays the recalled trace

Returns

`<Boolean>` 0,1

Examples

`MMEMory:EMC:RECAll:TRACe2:SHOW` might return `0`, indicating recalled trace 2 is hidden.

MMEMory:EMC:REPOrt:GENerate:FINAl (No Query Form)

Generates the report and saves the report file in the specified format under specified location in EMC display.

Conditions

Measurement view: EMC - EMI Display

Group

Mass Memory commands

Syntax

`MMEMory:EMC:REPOrt:GENerate:FINAl <file_name>`

Arguments

`<file name> = <String>` specifies the file and format to store the report.

Returns

None

Examples

`MMEMORY:EMC:REPORT:GENERATE:FINAL "C:\SignalVu-PC Files\report.pdf"` generates and stores the report in pdf format with file named report.pdf.

MMEMory:EMC:REPOrt:HISTory:DELete:ALL (No Query Form)

Deletes all the saved test results in history list of report setup in EMC display.

Conditions

Measurement view: EMC - EMI Display

Group

Mass Memory commands

Syntax

`MMEMory:EMC:REPOrt:HISTory:DELete:ALL`

Arguments

None

Returns

None

Examples

`MMEMORY:EMC:REPORT:HISTORY:DELETE:ALL` deletes all the saved test results in the history list of report setup.

MMEMory:EMC:REPOrt:SAVE:RUN (No Query Form)

Saves the test results and adds to the history list of report setup in EMC display.

Conditions

Measurement view: EMC - EMI Display

Group

Mass Memory commands

Syntax

`MMEMory:EMC:REPOrt:SAVE:RUN`

Arguments

None

Returns

None

Examples

`MMEMORY:EMC:REPORT:SAVE:RUN` saves the test results and adds to the history list.

MMEMory:EMC:SCAN:TRACe<x>:STORe (No Query Form)

Stores the specified live scan trace into the specified file in the EMC view.

<x> = <NR1> Trace number

Conditions

Measurement view: EMC - EMI Display

Group

Mass Memory commands

Syntax

`MMEMory:EMC:SCAN:TRACe<x>:STORe <file name>`

Arguments

`<file name> = <String>` specifies the file to store the scan trace with extension as .emcemi

Returns

None

Examples

`MMEMory:EMC:SCAN:TRACe1:STORe "C:\SignalVu-PC Files\sample.emcemi"` stores the scan trace 1 in the sample.emcemi file.

MMEMory:EMC:SPOT:TRACe<x>:STORe (No Query Form)

Stores the specified live spot trace into the specified file in the EMC view.

<x> = <NR1> Trace number

Conditions

Measurement view: EMC - EMI Display

Group

Mass Memory commands

Syntax

```
MMEMory:EMC:SPOT:TRACe<x>:STORe <file name>
```

Arguments

`<file name> = <String>` specifies the file to store the spot trace with extension as .emcemi

Returns

None

Examples

`MMEMory:EMC:SPOT:TRACe2:STORe "C:\SignalVu-PC Files\trace.emcemi"` stores the spot trace 2 in the trace.emcemi file.

[MMEMory:{FSETtling|PSETtling}:LOAD:TRACe<x>

Loads the Frequency or Phase Settling Time trace data from the specified file. The parameter <x> = 1 or 2; only Trace1 and Trace2 are used for settling time measurements.

Conditions

Measurement views: Frequency or Phase Settling

Group

Mass Memory commands

Syntax

```
[MMEMory:{FSETtling|PSETtling}:LOAD:TRACe<x> <file_name>
```

Arguments

`<file_name>::<string>` specifies the file from which to load trace data.

Examples

`MMEMORY:PSETTLING:LOAD:TRACE1 "Wfm1"` loads Trace1 data from the Wfm1 file.

[MMEMory:{FSETtling|PSETtling}:SHOW:TRACe<x>

Enables display of a recalled Frequency or Phase Settling trace file in Trace<x>. The result is the same as selecting “Show recalled trace” in the Trace control panel.

Conditions

Measurement views: Frequency and Phase Settling

Group

Mass Memory commands

Syntax

```
[MMEMory:{FSETtling|PSETtling}:SHOW:TRACe<x> { OFF | ON | 0 | 1 }
[MMEMory:{FSETtling|PSETtling}:SHOW:TRACe<x>?
```

Related Commands

[MMEMory:{FSETtling|PSETtling}:LOAD:TRACe<x> (on page 1507)

Arguments

OFF or 0 disables display of the recalled trace.

ON or 1 enables display of the recalled trace.

Examples

MMEMORY:PSETTLING:SHOW:TRACE1 ON enables display of the recalled data loaded in Trace1.

[MMEMory:{FSETtling|PSETtling}:STORe:TRACe<x>

Stores the Frequency or Phase Settling trace data in the specified file. The parameter <x> = 1 or 2; only Trace1 and Trace2 are used for Frequency and Phase Settling.

Conditions

Measurement views: Phase noise

Group

Mass Memory commands

Syntax

[MMEMory:{FSETtling|PSETtling}:STORe:TRACe<x> <file_name>

Arguments

<file_name>::<string> specifies the file in which to store trace data.

Examples

MMEMORY:PNOISE:STORE:TRACE1 "TRACE1" stores Trace 1 data in the TRACE1 file.

MMEMory:FVTime[:MEASview<y>]:LOAD:TRACe (No Query Form)

Loads the Frequency versus Time trace data from the specified file.

Conditions

Measurement views: Frequency versus Time

Group

Mass Memory commands

Syntax

MMEMory:FVTime[:MEASview<y>]:LOAD:TRACe <file_name>

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<file_name>::<string> specifies the file from which to load trace data.The file extension is .FreqVsTime. You can omit the extension.

Examples

`MMEMORY:FVTIME:MEASview1:LOAD:TRACE"Sample1"` loads the Frequency versus Time trace data from the *Sample1.FreqVsTime* file.

MMEMory:FVTIME[:MEASview<y>]:SHOW:TRACe<x>

Enables display of a recalled trace file in Trace<x>. The result is the same as selecting “Show recalled trace” in the Recall traces dialog box or selecting Show in the Trace Overview display.

Conditions

Measurement views: Frequency versus Time

Group

Mass Memory commands

Syntax

```
MMEMory:FVTIME[:MEASview<y>]:SHOW:TRACe<x> { OFF | ON | 0 | 1 }
MMEMory:FVTIME:SHOW:TRACe<x>?
```

Related Commands

MMEMory:FVTime[:MEASview<y>]:LOAD:TRACe (on page 1508)

Arguments

`OFF` or 0 disables display of the recalled trace.

`ON` or 1 enables display of the recalled trace.

Examples

`MMEMORY:FVTIME:MEASview1:SHOW:TRACE2 ON` enables display of the recalled data loaded in Trace2.

MMEMory:FVTime[:MEASview<y>]:STORe:TRACe (No Query Form)

Stores the Frequency versus Time trace data in the specified file.

Conditions

Measurement views: Frequency versus Time

Group

Mass Memory commands

Syntax

```
MMEMory:FVTime[:MEASview<y>]:STORe:TRACe <file_name>
```

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

`<file_name>::<string>` specifies the file in which to store trace data.The file extension is .FreqVsTime. You can omit the extension.

Examples

MMEMORY:FVTIME:MEASview1:STORE:TRACE"Sample1" stores the Frequency versus Time trace data in the *Sample1.FreqVsTime* file.

MMEMory:IQVTime[:MEASVIEW<y>]:LOAD:TRACe:I (No Query Form)

Loads I trace data from the specified file.

Conditions

Measurement views: RF I&Q versus Time

Group

Mass Memory commands

Syntax

MMEMory:IQVTime[:MEASVIEW<y>]:LOAD:TRACe:I <file_name>

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<file_name>::<string> specifies the file from which to load trace data.The file extension is .RFIQVsTime. You can omit the extension.

Examples

MMEMORY:IQVTIME:MEASVIEW1:LOAD:TRACE:I "Sample1" loads the I trace data from the *Sample1.RFIQVsTime* file.

MMEMory:IQVTime[:MEASVIEW<y>]:LOAD:TRACe:Q (No Query Form)

Loads Q trace data from the specified file.

Conditions

Measurement views: RF I&Q versus Time

Group

Mass Memory commands

Syntax

MMEMory:IQVTime[:MEASVIEW<y>]:LOAD:TRACe:Q <file_name>

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<file_name>::<string> specifies the file from which to load trace data.The file extension is .RFIQVsTime. You can omit the extension.

Examples

MMEMORY:IQVTIME:MEASVIEW1:LOAD:TRACE:Q "Sample2" loads the I trace data from the *Sample2.RFIQVsTime* file.

MMEMory:IQVTIME[:MEASVIEW<y>]:SHOW:TRACe:I

Enables display of a recalled trace file in Trace. The result is the same as selecting “Show recalled trace” in the Recall traces dialog box or selecting Show in the Trace Overview display.

Conditions

Measurement views: RF I&Q versus Time

Group

Mass Memory commands

Syntax

```
MMEMory:IQVTIME[:MEASVIEW<y>]:SHOW:TRACe:I { OFF | ON | 0 | 1 }
MMEMory:IQVTIME[:MEASVIEW<y>]:SHOW:TRACe:I?
```

Related Commands

MMEMory:IQVTime[:MEASVIEW<y>]:LOAD:TRACe:I (on page 1510)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

OFF or 0 disables display of the recalled trace.

ON or 1 enables display of the recalled trace.

Examples

MMEMORY:IQVTIME:MEASVIEW1:SHOW:TRACE:I ON enables display of the recalled data loaded in the Trace.

MMEMory:IQVTIME[:MEASVIEW<y>]:SHOW:TRACe<x>:Q

Enables display of a recalled trace file in Trace<x>. The result is the same as selecting “Show recalled trace” in the Recall traces dialog box or selecting Show in the Trace Overview display.

Conditions

Measurement views: RF I&Q versus Time

Group

Mass Memory commands

Syntax

```
MMEMory:IQVTIME[:MEASVIEW<y>]:SHOW:TRACe<x>:Q { OFF | ON | 0 | 1 }
MMEMory:IQVTIME[:MEASVIEW<y>]:SHOW:TRACe<x>:Q?
```

Related Commands

MMEMory:IQVTIME[:MEASVIEW<y>]:SHOW:TRACe<x>:Q (on page 1511)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

OFF or 0 disables display of the recalled trace.

ON or 1 enables display of the recalled trace.

Examples

MMEMORY:IQVTIME:MEASVIEW1:SHOW:TRACE3:Q ON enables display of the recalled data loaded in Trace3.

MMEMory:IQVTime[:MEASVIEW<y>]:STORe:TRACe:I (No Query Form)

Stores I trace data in the specified file.

Conditions

Measurement views: RF I&Q versus Time

Group

Mass Memory commands

Syntax

MMEMory:IQVTime[:MEASVIEW<y>]:STORe:TRACe:I <file_name>

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<file_name>::<string> specifies the file in which to store trace data.The file extension is .RFIQVsTime. You can omit the extension.

Examples

MMEMORY:IQVTIME:MEASVIEW1:STORE:TRACE:I"Sample1" stores the I trace data in the *Sample1.RFIQVsTime* file.

MMEMory:IQVTime[:MEASVIEW<y>]:STORe:TRACe:Q (No Query Form)

Stores Q trace data in the specified file.

Conditions

Measurement views: RF I&Q versus Time

Group

Mass Memory commands

Syntax

MMEMory:IQVTime[:MEASVIEW<y>]:STORe:TRACe:Q <file_name>

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<file_name>::<string> specifies the file in which to store trace data.The file extension is .RFIQVsTime. You can omit the extension.

Examples

MMEMORY:IQVTIME:MEASVIEW1:STORE:TRACE:Q "Sample2" stores the Q trace data in the *Sample2.RFIQVsTime* file.

MMEMory:LOAD:IQ (No Query Form)

Loads time domain IQ waveform into the acquisition memory from a file.

Conditions

Measurement views: All

Group

Mass Memory commands

Syntax

`MMEMory:LOAD:IQ <file_name>`

Arguments

`<file_name>::<string>` specifies the file to load IQ data from. The file extension .tiq is for non-multichannel tiq files and .tiqm for multichannel tiq files. You can omit the extension.

Examples

`MMEMORY:LOAD:IQ "IQ1"` loads IQ data from the *IQ1.tiq* file.

MMEMory:LOAD:STATe (No Query Form)

Loads the instrument setup from a specified file for the currently selected view.

Conditions

Measurement views: All

Group

Mass Memory commands

Syntax

`MMEMory:LOAD:STATe <file_name>`

Arguments

`<file_name>::<string>` specifies the file to load the instrument setup from.The file extension is .setup. You can omit the extension.

Examples

`MMEMORY:LOAD:STATE "STATE1"` loads the instrument setup from the *STATE1.setup* file.

MMEMory:LOAD:TRACe (No Query Form)

Loads the trace data from a specified file for the currently selected view.

Conditions

Measurement views (See the table below)

Group

Mass Memory commands

Syntax

```
MMEMory:LOAD:TRACe <file_name>
```

Related Commands

MMEMory:STORe:TRACe (on page 1537)

Arguments

`<file_name>::<string>` specifies the file to load the trace data from. The file extension is named for the measurement view as shown in the following table.The trace file is not available in some views which are indicated by "NA" in the file extension column.

For the views that allow multiple-trace display, select the trace to load or store using the command listed in the trace selection column in the following table. You can load the specified trace with any trace data that you saved in the view if the traces have the same units. For example, you can load Trace 2 with Trace 1 data that you saved.

Trace file extension and trace selection command

Display group	Measurement view	File extension	Trace selection
General signal viewing	Spectrum	Specan	TRACe<x>:SPECtrum:SELect
	DPX spectrum	dpt	TRACe<x>:DPX:SELect
	Amplitude versus Time	AmplVsTime	TRACe<x>:AVTime:SELect
	Frequency versus Time	FreqVsTime	NA
	Phase versus Time	PhaseVsTime	NA
	RF I&Q versus Time	RFIQVsTime	NA
	Spectrogram	Sogram	NA
	Time overview	NA	NA
General purpose analog modulation (Option 21 only)	AM	AmVsTime	NA
	FM	FmVsTime	NA
	PM	PmVsTime	NA

Display group	Measurement view	File extension	Trace selection
General purpose digital modulation (Option 21 only)	Constellation	NA	NA
	Demodulated I & Q versus Time	NA	NA
	Eye Diagram	NA	NA
	EVM versus Time	NA	NA
	Frequency Deviation versus Time	NA	NA
	Magnitude error versus Time	NA	NA
	Phase error versus Time	NA	NA
	Signal quality	NA	NA
	Symbol table	NA	NA
	Trellis Diagram	NA	NA
RF measurements	CCDF	CCDF	TRACe<x>:CCDF:SELect
	Channel power and ACPR	NA	NA
	MCPR	NA	NA
	Occupied Bandwidth	NA	NA
	Phase noise (Option 11 only)	PhaseNoise	TRACe<x>: {FSETtling|PSETtling}:SELect
	Spurious	NA	NA
Pulsed RF	Pulse statistics	NA	NA
	Pulse table	NA	NA
	Pulse trace	NA	NA

Examples

`MMEMORY:LOAD:TRACE "TRACE1"` loads the trace data from the `TRACE1.Specan` file when the spectrum view is selected.

MMEMory:LTE:ACLR:LOAD:TABLe (No Query Form)

Loads the LTE ACLR table containing the limits for enabled ranges from the specified .cvs file.

Conditions

Measurement view: LTE ACLR

Group

Mass Memory commands

Syntax

`MMEMory:LTE:ACLR:LOAD:TABLe <string>`

Arguments

`<file_name>::<string>` specifies the file from which to load the LTE ACLR table. The file extension is `.csv`. You can omit the extension.

Examples

`MMEMORY:LTE:ACLR:LOAD:TABLE "Table1"` loads the LTE ACLR table from the *Table1.csv* file.

MMEMory:LTE:ACLR:STORe:TABLe (No Query Form)

Stores the LTE ACLR table containing the limits for enabled ranges in a specified

.cvs file, allowing you to export the file into Microsoft Excel or other database system.

Conditions

Measurement view: LTE ACLR

Group

Mass Memory commands

Syntax

`MMEMory:LTE:ACLR:STORe:TABLe <string>`

Arguments

`<file_name>::<string>` specifies the file in which to store the LTE ACLR table.

The file extension is *.csv*. You can omit the extension.

Examples

`MMEMORY:LTE:ACLR:STORE:TABLE "Table1"` stores the LTE ACLR table in the Table1.csv file.

MMEMory:MAPit:LOAD (No Query Form)

Loads the MapIt measurement results from the specified file.

Conditions

Measurement views: Spectrum, DPX, Spectrogram, Amplitude vs Time, Channel Power, Occupied Bandwidth

Group

Mass Memory commands

Syntax

`MMEMory:MAPit:LOAD <filename>`

Arguments

`<filename>::<string>` the name of the file from which to load the MapIt data.

Examples

`MMEMORY:MAPIT:LOAD "C:\SignalVu-PC files\MapIt_results1"` Loads the MapIt measurement results in the MapIt_results1 file from the C:SignalVu-PC files folder.

MMEMory:MAPit:STORe (No Query Form)

Save the selected MapIt measurement results to a file. The results to save are selected by the <result_index>. The saved files are compressed .zip files.

Conditions

Measurement views: Spectrum, DPX, Spectrogram, Amplitude vs Time, Channel Power, Occupied Bandwidth

Group

Mass Memory commands

```
Syntax
MMEMory:MAPit:STORe <filename>, <result_index> [,<result_index]
```

Arguments

`<filenam>::<string>` the name of the file in which to store the MapIt measurement results.

`<result_index>::<NRf>`**.** The 1-based index of the MapIt result to load. See the MapIt editor for the list of results. index 1 is the first result at the top of the list.

Examples

```
MMEMORY:MAPIT:STORE ??
```

MMEMory:MAPit:STORe:ALL (No Query Form)

Saves all of the MapIt measurement results from memory to the specified file.

Conditions

Measurement views: Spectrum, DPX, Spectrogram, Amplitude vs Time, Channel Power, Occupied Bandwidth

Group

Mass Memory commands

Syntax

```
MMEMory:MAPit:STORe:ALL <filename>
```

Arguments

`<filename>::<string>` the name of the file in which to store the MapIt measurement results.

Examples

`MMEMORY:MAPIT:STORE:ALL "C:\SignalVu-PC files\MapIt_results1"` stores the MapIt measurement results in the MapIt_results1 file in the C:SignalVu-PC files folder.

MMEMory:NOISe:GAIN:LOAD:TRACe (No Query Form)

Loads the Noise Gain trace data from the specified file.

Conditions

Measurement view: Gain

Group

Mass Memory commands

Syntax

```
MMEMory:NOISe:GAIN:LOAD:TRACe <file_name>
```

Related Commands

MMEMory:NOISe:GAIN:STORe:TRACe (on page 1518)

Arguments

`<file_name>` = the file from which to load trace data.

Examples

`MMEMORY:NOISE:GAIN:LOAD:TRACE trace1` will load the trace1 data file.

MMEMory:NOISe:GAIN:SHOW:TRACe

Show or hide the Noise Gain trace from the specified file.

Conditions

Measurement view: Gain display

Group

Mass Memory commands

Syntax

```
MMEMory:NOISe:GAIN:SHOW:TRACe { OFF | ON | 0 | 1 }
MMEMory:NOISe:GAIN:SHOW:TRACe
```

Arguments

`OFF` or 0 sets the trace to hide from the specified file.

`ON` or 1 sets the trace to show from the specified file.

Returns

`0` means the trace is hidden.

`1` means the trace is showing.

Examples

`MMEMORY:NOISE:GAIN:SHOW:TRACE?` might return `1`, indicating that the trace is set to show from the specified file.

MMEMory:NOISe:GAIN:STORe:TRACe (No Query Form)

Stores the Noise Gain trace data in the specified file.

Conditions

Measurement view: Gain display

Group

Mass Memory commands

Syntax

```
MMEMory:NOISe:GAIN:STORe:TRACe <file_name>
```

Related Commands

MMEMory:NOISe:GAIN:LOAD:TRACe (on page 1517)

Arguments

`<file_name>` = the file in which to store trace data.

Examples

```
MMEMORY:NOISE:GAIN:STORE:TRACE
```

MMEMory:Noise:LOAD:ENRCALibration (No Query Form)

Loads the specified Noise Figure ENR table.

Conditions

Measurement view: Noise display

Group

Mass Memory commands

Syntax

```
MMEMory:Noise:LOAD:ENRCALibration <filename>
```

Arguments

`<file_name>` = the file from which to load ENR calibration data.

Examples

```
MMEMORY:NOISE:LOAD:ENRCALIBRATION
```

MMEMory:Noise:LOAD:FREQTABle (No Query Form)

Loads the specified Noise Figure Frequency table.

Conditions

Measurement view: Noise display

Group

Mass Memory commands

Syntax

```
MMEMory:Noise:LOAD:FREQTABle <filename>
```

Arguments

`<filename>` = the file from which to load the frequency table data.

Examples

```
MMEMORY:NOISE:LOAD:FREQTABLE
```

MMEMory:Noise:STORe:ENRCALibration (No Query Form)

Saves the Noise Figure ENR table to the specified location.

Conditions

Measurement view: Noise display

Group

Mass Memory commands

Syntax

```
MMEMory:Noise:STORe:ENRCALibration <filename>
```

Arguments

`<filename>` = the file in which to store ENR calibration data.

Examples

```
MMEMORY:NOISE:STORE:ENRCALIBRATION
```

MMEMory:Noise:STORe:FREQTABle (No Query Form)

Saves the Noise Figure Frequency table to the specified location.

Conditions

Measurement view: Noise display

Group

Mass Memory commands

Syntax

```
MMEMory:Noise:STORe:FREQTABle <filename>
```

Arguments

`<filename>` = the file in which to store the frequency table data.

Examples

```
MMEMORY:NOISE:STORE:FREQTABLE
```

MMEMory:OFDM:CONSte:LOAD:TRACe (No Query Form)

Loads the OFDM constellation trace data from the specified file.

Conditions

Measurement views: OFDM

Group

Mass Memory commands

Syntax

```
MMEMory:OFDM:CONSte:LOAD:TRACe <file_name>
```

Arguments

`<file_name>::<string>` specifies the file from which to load the trace data.

Examples

`MMEMORY:OFDM:CONSTE:LOAD:TRACE "Sample1"` loads the trace data from the file with the OFDM Constellation measurement.

MMEMory:OFDM:CONSTe:SHOW:TRACe

Enables the display of a recalled trace.

Conditions

Measurement views: OFDM

Group

Mass Memory commands

Syntax

```
MMEMory:OFDM:CONSTe:SHOW:TRACe{ OFF | ON | 0 | 1 }
MMEMory:OFDM:CONSTe:SHOW:TRACe?
```

Arguments

`OFF` or 0 disables display of the recalled trace.

`ON` or 1 enables display of the recalled trace.

Examples

`MMEMORY:OFDM:CONSTE:SHOW:TRACE ON` enables the display of the recalled data loaded in the trace.

MMEMory:OFDM:CONSte:STORe:TRACe (No Query Form)

Stores the OFDM constellation measurement trace data in the specified file.

Conditions

Measurement views: OFDM

Group

Mass Memory commands

Syntax

```
MMEMory:OFDM:CONSte:STORe:TRACe <file_name>
```

Arguments

`<file_name>::<string>` specifies the file in which to store the trace data.

Examples

`MMEMORY:OFDM:CONSTE:STORE:TRACE "Sample1"` stores the trace data in the Sample1 file in the OFDM Constellation measurement.

MMEMory:PHVTime[:MEASVIEW<y>]:LOAD:TRACe (No Query Form)

Loads the Phase versus Time trace data from the specified file.

Conditions

Measurement views: Phase versus Time

Group

Mass Memory commands

Syntax

```
MMEMory:PHVTime[:MEASVIEW<y>]:LOAD:TRACe <file_name>
```

Arguments

`<file_name>::<string>` specifies the file from which to load trace data.The file extension is .PhaseVsTime. You can omit the extension.

Examples

`MMEMORY:PHVTIME:MEASVIEW1:LOAD:TRACE "Sample1"` loads the Phase versus Time trace data from the *Sample1.PhaseVsTime* file.

MMEMory:PHVTime[:MEASview<y>]:SHOW:TRACe

Enables display of a recalled trace file in Trace<x>. The result is the same as selecting “Show recalled trace” in the Recall traces dialog box or selecting Show in the Trace Overview display.

Conditions

Measurement views: Phase versus Time

Group

Mass Memory commands

Syntax

```
MMEMory:PHVTime[:MEASview<y>]:SHOW:TRACe { OFF | ON | 0 | 1 }
MMEMory:PHVTime:SHOW:TRACe?
```

Related Commands

MMEMory:PHVTime[:MEASview<y>]:LOAD:TRACe (on page 1522)

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

`OFF` or 0 disables display of the recalled trace.

`ON` or 1 enables display of the recalled trace.

Examples

`MMEMORY:PHVTIME:MEASview1:SHOW:TRACE ON` enables display of the recalled data loaded in the Trace.

MMEMory:PHVTime[:MEASVIEW<y>]:STORe:TRACe (No Query Form)

Stores the Phase versus Time trace data in the specified file.

Conditions

Measurement views: Phase versus Time

Group

Mass Memory commands

Syntax

```
MMEMory:PHVTime[:MEASVIEW<y>]:STORe:TRACe <file_name>
```

Arguments

`<file_name>::<string>` specifies the file in which to store trace data.The file extension is .PhaseVsTime. You can omit the extension.

Examples

`MMEMORY:PHVTIME:MEASVIEW1:STORE:TRACE "Sample1"` stores the Phase versus Time trace data in the *Sample1.PhaseVsTime* file.

MMEMory:PNOise:LOAD:TRACe<x> (No Query Form)

Loads the Phase Noise trace data from the specified file. The parameter <x> = 1 or 2; only Trace1 and Trace2 are used for Phase Noise measurement.

Conditions

Measurement views: Phase noise

Group

Mass Memory commands

Syntax

```
MMEMory:PNOise:LOAD:TRACe<x> <file_name>
```

Arguments

`<file_name>::<string>` specifies the file from which to load trace data.

Examples

`MMEMORY:PNOISE:LOAD:TRACE1 "Wfm1"` loads Trace1 data from the Wfm1 file.

MMEMory:PNOise:SHOW:TRACe<x>

Enables display of a recalled Phase Noise trace file in Trace<x>. The result is the same as selecting “Show recalled trace” in the Recall traces dialog box or selecting Show in the Trace Overview display.

Conditions

Measurement views: Phase noise

Group

Mass Memory commands

Syntax

```
MMEMory:PNOise:SHOW:TRACe<x> { OFF | ON | 0 | 1 }
MMEMory:PNOise:SHOW:TRACe<x>?
```

Related Commands

MMEMory:PNOise:LOAD:TRACe<x> (on page 1523)

Arguments

`OFF` or 0 disables display of the recalled trace.

`ON` or 1 enables display of the recalled trace.

Examples

`MMEMORY:PNOISE:SHOW:TRACE1 ON` enables display of the recalled data loaded in Trace1.

MMEMory:PNOise:STORe:TRACe<x> (No Query Form)

Stores the Phase Noise trace data in the specified file. The parameter <x> = 1 or 2; only Trace1 and Trace2 are used for Phase Noise measurement.

Conditions

Measurement views: Phase noise

Group

Mass Memory commands

Syntax

```
MMEMory:PNOise:STORe:TRACe<x> <file_name>
```

Arguments

`<file_name>::<string>` specifies the file in which to store trace data.

Examples

`MMEMORY:PNOISE:STORE:TRACE1 "WFM1"` stores Trace 1 data in the WFM1 file.

MMEMory:SEM:LOAD:TABLe (No Query Form)

Loads the Spectral Emissions Mask table containing the limits for enabled ranges from the specified file.

Conditions

Measurement views: Spectral Emissions Mask

Group

Mass Memory commands

Syntax

```
MMEMory:SEM:LOAD:TABLe <file_name>
```

Arguments

<file_name>::<string> specifies the file to load the SEM table from.The file extension is **.csv**. You can omit the extension.

Examples

MMEMORY:SEM:LOAD:TABLE "Table1" loads the SEM table from the *Table1.csv* file.

MMEMory:SEM:STORe:TABLe (No Query Form)

Stores the Spectral Emissions Mask table containing the limits for enabled ranges in a specified file in the CSV format, allowing you to export the file into Microsoft Excel or other database systems.

Conditions

Measurement views: Spectral Emissions Mask

Group

Mass Memory commands

Syntax

MMEMory:SEM:STORe:TABLe <file_name>

Arguments

<file_name>::<string> specifies the file to store the SEM table in.The file extension is .csv. You can omit the extension.

Examples

MMEMORY:SEM:STORE:TABLE "Table1" stores the SEM table in the *Table1.csv* file.

MMEMory:SGRam[:MEASview<y>]:LOAD:TRACe (No Query Form)

Loads the Spectrogram trace data from the specified file. No trace number is needed or allowed.

Conditions

Measurement views: Spectrogram

Group

Mass Memory commands

Syntax

MMEMory:SGRam[:MEASview<y>]:LOAD:TRACe <file_name>

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

<file_name>::<string> specifies the file from which to load trace data.The file extension is .sogram. You can omit the extension.

Examples

MMEMory:SGRam:MEASview1:LOAD:TRACe5 "Spec2" loads Trace5 data from the file *Spec2.sogram*.

MMEMory:SGRam[:MEASview<y>]:SHOW:TRACe

Enables display of a recalled trace file in Trace. The result is the same as selecting “Show recalled trace” in the Recall traces dialog box or selecting Show in the Trace Overview display.

Conditions

Measurement views: Spectrogram

Group

Mass Memory commands

Syntax

```
MMEMory:SGRam[:MEASview<y>]:SHOW:TRACe { OFF | ON | 0 | 1 }
MMEMory:SGRam[:MEASview<y>]:SHOW:TRACe?
```

Related Commands

MMEMory:SGRam[:MEASview<y>]:LOAD:TRACe (on page 1525)

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

OFF or 0 disables display of the recalled trace.

ON or 1 enables display of the recalled trace.

Returns

{ OFF | ON | 0 | 1 }

Examples

MMEMory:SGRam:MEASview1:SHOW:TRACe ON enables display of the recalled data loaded in Trace.

MMEMory:SGRam[:MEASview<y>]:STORe:TRACe (No Query Form)

Stores the Spectrogram trace data in the specified file. No trace number is needed or allowed.

NOTE: Only valid when the spectrum or spectrogram measurements are running.

Conditions

Measurement views: Spectrogram

Group

Mass Memory commands

Syntax

```
MMEMory:SGRam[:MEASview<y>]:STORe:TRACe <file_name>
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

<file_name>::<string> specifies the file in which to store trace data.The file extension is .sogram. You can omit the extension.

Examples

`MMEMory:SGRam:MEASview1:STORe:TRACe "Spec2"` stores Trace data in the *Spec2.sogram* file.

MMEMory:SGRam[:MEASview<y>]:STORe:PLOT

Stores the spectrogram plot data into a specified location and file.

Conditions

Measurement views: Spectrogram

Group

Mass Memory commands

Syntax

`MMEMory:SGRam[:MEASview<y>]:STORe:PLOT <file_name>`

Arguments

`<y>` = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

`<file_name>` = <String> specifies the file location and file name to which the plot to be saved.

Returns

None

Examples

`MMEMory:SGRam:MEASview1:STORe:PLOT "C:\SignalVu-PC Files\Sample1.Specan"` saves the plot in Sample1.Specan file.

MMEMory:SGRam[:MEASview<y>]:STORe:PLOT:DONe (Query only)

Determines whether the spectrogram plot data is saved or not saved.

Conditions

Measurement views: Spectrogram

Group

Mass Memory commands

Syntax

`MMEMory:SGRam[:MEASview<y>]:STORe:PLOT:DONe?`

Arguments

`<y>` = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Returns

`<Boolean>` 0, 1

`1` indicates the spectrogram plot data is saved.

`0` indicates the spectrogram plot data is not saved.

Examples

`MMEMory:SGRam:MEASview1:STORe:PLOT:DONe?` might return 1, indicating the spectrogram plot data is saved.

MMEMory:SPECtrum:MEASview<y>:LOAD:TRACe (No Query Form)

Loads the spectrum trace data from the specified file.

The parameter <x> = 1 to 4; Trace 5 (spectrogram) is not valid.

Conditions

Measurement views: Spectrum

Group

Mass Memory commands

Syntax

```
MMEMory:SPECtrum:MEASview<y>:LOAD:TRACe <file_name>
```

Arguments

`<file_name>::<string>` specifies the file from which to load trace data. The file extension is .Specan. You can omit the extension.

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`MMEMory:SPECtrum:MEASview2:LOAD:TRACe "C:\SignalVu-PC Files\Sample1.Specan"` loads Trace 1 data from Sample1.Specan file

MMEMory:SPECtrum[:MEASview<y>]:SHOW:TRACe<x>

Enables display of a recalled trace file in Trace<x>. The result is the same as selecting “Show recalled trace” in the Recall traces dialog box or selecting Show in the Trace Overview display.

Conditions

Measurement views: Spectrum

Group

Mass Memory commands

Syntax

```
MMEMory:SPECtrum[:MEASview<y>]:SHOW:TRACe<x> { OFF | ON | 0 | 1 }
MMEMory:SPECtrum[:MEASview<y>]:SHOW:TRACe<x>?
```

Related Commands

MMEMory:SPECtrum[:MEASview<y>]:LOAD:TRACe (on page 1528)

Arguments

`OFF` or `0` disables display of the recalled trace.

`ON` or `1` enables display of the recalled trace.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`MMEMORY:SPECTRUM:MEASVIEW2:SHOW:TRACE2 ON` enables display of the recalled data loaded in Trace2.

MMEMory:SPECtrum[:MEASview<y>]:STORe:PLOT (No Query Form)

Stores the spectrum plot data into a specified location and file.

Conditions

Measurement view: Spectrum

Group

Mass Memory commands

Syntax

`MMEMory:SPECtrum[:MEASview<y>]:STORe:PLOT <file_name>`

Arguments

`<file_name> = <String>` specifies the file location and file name to which the plot to be saved.

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

None

Examples

`MMEMORY:SPECTRUM:MEASVIEW2:STORE:PLOT "C:\SignalVu-PC Files\Sample1.Specan"` saves the plot in Sample1.Specan file

MMEMory:SPECtrum[:MEASview<y>]:STORe:PLOT:DONe (Query Only)

Determines whether the spectrum plot data is saved or not saved.

Conditions

Measurement view: Spectrum

Group

Mass Memory commands

Syntax

`MMEMory:SPECtrum[:MEASview<y>]:STORe:PLOT:DONe?`

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

<Boolean> 0, 1

1 indicates the spectrum plot data is saved

0 indicates the spectrum plot data is not saved

Examples

`MMEMORY:SPECTRUM:MEASVIEW2:STORE:PLOT:DONE` might return `1`, indicating the spectrum plot data is saved

MMEMory:SPECtrum[:MEASview<y>]:STORe:TRACe (No Query Form)

Stores the spectrum trace data in the specified file. The parameter <x> = 1 to 5; All traces are valid.

NOTE: TRACe5 (spectrogram) is valid when the spectrum and spectrogram measurements are running.

Conditions

Measurement views: Spectrum

Group

Mass Memory commands

Syntax

`MMEMory:SPECtrum[:MEASview<y>]:STORe:TRACe <file_name>`

Arguments

`<file_name>::<string>` specifies the file in which to store trace data.The file extension is .Specan. You can omit the extension.

The file extension is named for the measurement view as shown in the table (Click here for Trace file extension and trace selection command) The trace file is not available in some views which are indicated by "NA" in the file extension column. For the views that allow multiple-trace display, select the trace to load or store using the command listed in the trace selection column.

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`MMEMORY:SPECTRUM:MEASview2:STORE:TRACE "TRACE1"` stores the trace data in the TRACE1.Specan file when the spectrum view is selected.

MMEMory:SPURious:LOAD:TABLe (No Query Form)

Loads the spurious table containing the limits for enabled ranges from the specified file.

Conditions

Measurement views: Spurious

Group

Mass Memory commands

Syntax

`MMEMory:SPURious:LOAD:TABLe <file_name>`

Arguments

`<file_name>::<string>` specifies the file to load the spurious table from.The file extension is `.csv`. You can omit the extension.

Examples

`MMEMORY:SPURIOUS:LOAD:TRACE1 "Table1"` loads the spurious table from the *Table1.csv* file.

MMEMory:SPURious:RECAll:TRACe (No Query Form)

Recalls the spurious trace data from the specified file and trace number.

Conditions

Measurement view: Spurious Display

Group

Mass Memory commands

Syntax

`MMEMory:SPURious:RECAll:TRACe <x> <file_name>`

Arguments

<x> = 1,2,3,4 is the trace `<file_name> = <String>` specifies the file location and file name to which the trace data to be Recalled

Returns

None

Examples

`MMEMORY:SPURIOUS:RECALL:TRACE "C:\SignalVu-PC Files\Sample1.spurious"` recalls the Trace 3 data from the Sample1.spurious file.

MMEMory:SPURious:SHOW:TRACe (No Query Form)

Displays the spurious trace data from the specified file and trace number.

Conditions

Measurement view: Spurious Display

Group

Mass Memory commands

Syntax

`MMEMory:SPURious:SHOW:TRACe <x> <file_name>`

Arguments

`<x>` = 1,2,3,4 is the trace

`<file_name> = <String>` specifies the file location and file name to which the trace data to be displayed.

Returns

None

Examples

`MMEMORY:SPURIOUS:SHOW:TRACE "C:\SignalVu-PC Files\Sample1.spurious"` displays the Trace 2 data from Sample1.spurious file.

MMEMory:SPURious:STORe:TABLe (No Query Form)

Stores the spurious table containing the limits for enabled ranges in a specified file in the CSV (Comma Separated Values) format, allowing you to export the file into Microsoft Excel or other database systems.

Conditions

Measurement views: Spurious

Group

Mass Memory commands

Syntax

`MMEMory:SPURious:STORe:TABLe <file_name>`

Arguments

`<file_name>::<string>` specifies the file to store the spurious table in.The file extension is .csv. You can omit the extension.

Examples

`MMEMORY:SPURIOUS:STORE:TABLE "Table1"` stores the spurious table in the *Table1.csv* file.

MMEMory:SPURious:STORe:TRACe (No Query Form)

Stores the specified spurious trace data into a specified location and file.

Conditions

Measurement view: Spurious Display

Group

Mass Memory commands

Syntax

`MMEMory:SPURious:STORe:TRACe <x> <file_name>`

Arguments

<x> = 1,2,3,4 is the trace

`<file_name> = <String>` specifies the file location and file name to which the trace data to be saved.

Returns

None

Examples

`MMEMORY:SPURIOUS:STORE:TRACE "C:\SignalVu-PC Files\Sample1.spurious"` saves the Trace 1 data in Sample1.spurious file.

MMEMory:STORe:IQ (No Query Form)

Saves time-domain IQ waveform in the acquisition memory to a specified file.

Conditions

Measurement views: All

Group

Mass Memory commands

Syntax

`MMEMory:STORe:IQ <file_name>`

Arguments

`<file_name>::<string>` specifies the file to save IQ data.

The file extension .tiq is for non-multichannel setup and .tiqm for multichannel setup. You can omit the extension.

Examples

`MMEMORY:STORE:IQ "IQ1"` saves IQ data to the *IQ1.tiq* file.

MMEMory:STORe:IQ:CSV (No Query Form)

Saves time-domain IQ waveform in the acquisition memory to a specified file in the CSV (Comma Separated Values) format, allowing you to export the file into Microsoft Excel or other database systems.

Conditions

Measurement views: All

Group

Mass Memory commands

Syntax

`MMEMory:STORe:IQ:CSV <file_name>`

Arguments

`<file_name>::<string>` specifies the file to save IQ data.The file extension is .csv. You can omit the extension.

Examples

`MMEMORY:STORE:IQ:CSV "IQ2"` saves IQ data to the *IQ2.cs(less t eq)* file.

MMEMory:STORe:IQ:MAT (No Query Form)

Saves time-domain IQ waveform in the acquisition memory to a specified file in the MATLAB format, allowing you to export the file into the MATLAB technical computing environment.

Conditions

Measurement views: All

Group

Mass Memory commands

Syntax

`MMEMory:STORe:IQ:MAT <file_name>`

Arguments

`<file_name>::<string>` specifies the file to save IQ data.The file extension is .mat. You can omit the extension.

Examples

`MMEMORY:STORE:IQ:MAT "IQ3"` saves IQ data to the *IQ3.mat* file.

MMEMory:STORe:IQ:SELEct:DATA

Sets or queries the data type stored when saving acquisition data files.

Conditions

Measurement views

Group

Mass Memory commands

Syntax

`MMEMory:STORe:IQ:SELEct:DATA { IQ | SPECtra | ALL }`

NOTE: Although the command includes IQ as part of the command, the command applies to all the data stored in the file, not only IQ data.

Arguments

`IQ` specifies that IQ data be saved in the data file.

`SPECtra` specifies that DPX spectra be saved in the data file.

`ALL` specifies that both IQ and DPX spectra be saved in the data file.

Examples

`MMEMORY:STORE:IQ:SELECT:DATA SPECTRA` sets the data type to DPX Spectra.

MMEMory:STORe:IQ:SELEct:FRAMes

Sets or queries which frames are saved when saving TIQ acquisition data files.

Conditions

Measurement views

Group

Mass Memory commands

Syntax

`MMEMory:STORe:IQ:SELEct:FRAMes { ACQuisition | HISTory | SELected | ALL }`

Arguments

`ACQuisition` saves the entire acquisition that contains the current frame.

`HISTory` saves the current frame.

SELected saves only selected frames.

ALL saves all acquired frames.

MMEMory:STORe:IQ:SELEct:LENGth

Sets or queries what portion of each record is saved when saving acquisition data.

Conditions

Measurement views

Group

Mass Memory commands

Syntax

```
MMEMory:STORe:IQ:SELEct:LENGth { ENTire | LENGth }
MMEMory:STORe:IQ:SELEct:LENGth?
```

Arguments

ENTire saves the entire record

LENGth saves the portion of the record being analyzed.

MMEMory:STORe:MSTate (No Query Form)

Stores the measurement parameters to a specified file in the ASCII text format for the currently selected view, allowing you to export the file into other applications.

Conditions

Measurement views: All

Group

Mass Memory commands

Syntax

```
MMEMory:STORe:MSTate <file_name>
```

Arguments

<file_name>::<string> specifies the file to store the measurement parameters. The file extension is .txt. You can omit the extension.

Examples

MMEMORY:STORE:MSTATE "MSTATE1" stores the measurement parameters to the *MSTATE1.txt* file.

MMEMory:STORe:RESults (No Query Form)

Stores the measurement results including measurement parameters and trace data to a specified file in the CSV (Comma Separated Values) format for the currently selected view, allowing you to export the file into Microsoft Excel or other database systems.

Conditions

Measurement views: All

Group

Mass Memory commands

Syntax

```
MMEMory:STORe:RESults <file_name>
```

Arguments

`<file_name>::<string>` specifies the file to store the measurement results.

The file extension is *.csv*. You can omit the extension.

Examples

`MMEMORY:STORE:RESULTS "RESULT1"` stores the measurement results to the *RESULT1.csv* file.

MMEMory:STORe:SCReen (No Query Form)

Stores the current display as a bitmap image file in one of several standard formats. When no format is specified, the system uses the default, .png.

Conditions

Measurement views: All

Group

Mass Memory commands

```
Syntax
MMEMory:STORe:SCReen <file_name>{ .bmp | .jpg | .png }
```

Arguments

`<file_name>::<string>` specifies the file to store the screen image.

Examples

`MMEMORY:STORE:SCREEN "RESULT1.png"` stores the current screen image into the file*RESULT1.png*.

MMEMory:STORe:STATe (No Query Form)

Stores the instrument setup to a specified file for the currently selected view.

Conditions

Measurement views: All

Group

Mass Memory commands

Syntax

```
MMEMory:STORe:STATe <file_name>
```

Arguments

`<file_name>::<string>` specifies the file to store the instrument setup.The file extension is setup. You can omit the extension.

Examples

`MMEMORY:STORE:STATE "STATE1"` stores the instrument setup in the *STATE1.setup* file.

MMEMory:STORe:TRACe (No Query Form)

Stores the trace data in a specified file for the currently selected view.

Conditions

Measurement views (See MMEMory:LOAD:TRACe (on page 1513).)

Group

Mass Memory commands

Syntax

`MMEMory:STORe:TRACe <file_name>`

Related Commands

MMEMory:IQVTIME:SHOW:TRACe:I (on page 1510)

Arguments

`<file_name>::<string>` specifies the file to store the trace data in.

The file extension is named for the measurement view as shown in the table in `MMEMory:LOAD:TRACe`. The trace file is not available in some views which are indicated by "NA" in the file extension column. For the views that allow multiple-trace display, select the trace to load or store using the command listed in the trace selection column.

Examples

`MMEMORY:STORE:TRACE "TRACE1"` stores the trace data in the *TRACE1.Specan* file when the spectrum view is selected.

MMEMory:TOVerview[:MEASview<y>]:LOAD:TRACe1 (No Query Form)

Loads the trace data from a specified file into Trace1. Refer to MMEMory:LOAD:TRACe (on page 1513) for the correct file extensions for each trace type.

Conditions

Measurement views

Group

Mass Memory commands

Syntax

`MMEMory:TOVerview[:MEASview<y>]:LOAD:TRACe1 <file_name>`

Related Commands

MMEMory[:MEASview<y>]:STORe:TRACe (on page 1537)

Arguments

`<file_name>::<string>` specifies the file path and name to load the trace data from. The file extension is named for the measurement view. See the table in `MMEMory:LOAD:TRACe`.

Most trace file types are compatible with only one Measurement view.

Examples

`MMEMORY:TOVERVIEW:MEASVIEW1:LOAD:TRACE1 "TrialB"` loads the trace data from the file *TRIALB.TOV* when the spectrum view is selected.

MMEMory:TOVerview[:MEASview<y>]:SHOW:TRACe1

Enables display of a recalled trace file in Trace<x>. The result is the same as selecting “Show recalled trace” in the Recall traces dialog box or selecting Show in the Trace Overview display.

Conditions

Measurement views

Group

Mass Memory commands

Syntax

```
MMEMory:TOVerview[:MEASview<y>:SHOW]:TRACe1 { OFF | ON | 0 | 1 }
MMEMory:TOVerview[:MEASview<y>]:SHOW:TRACe1?
```

Related Commands

MMEMory:TOVerview[:MEASview<y>]:LOAD:TRACe1 (on page 1537)

Arguments

`OFF` or `0` disables display of the recalled trace.

`ON` or `1` enables display of the recalled trace.

Examples

`MMEMORY:TOVERVIEW:MEASVIEW1:SHOW:TRACE1 ON` enables display of the recalled data loaded in Trace2.

MMEMory:TOVerview[:MEASview<y>]:STORe:TRACe1 (No Query Form)

Stores the trace data for Trace1 into the specified file.

Conditions

Measurement views

Group

Mass Memory commands

Syntax

```
MMEMory:TOVerview[:MEASview<y>]:STORe:TRACe1 <file_name>
```

Related Commands

MMEMory[:MEASview<y>]:STORe:TRACe (on page 1537)

Arguments

`<file_name>::<string>` specifies the file to store the trace data in.

The file extension is named for the measurement view as shown in the table (See MMEMory:LOAD:TRACe (on page 1513).) The trace file is not available in some views which are indicated by "NA" in the file extension column. For the views that allow multiple-trace display, select the trace to load or store using the command listed in the trace selection column.

Examples

`MMEMORY:TOVERVIEW:MEASVIEW1:STORE:TRACE1 "TrialB"` stores the Trace1 data into the file *TRIALB.TOV* when a spectrum view is selected.

MMEMory:TXGain:LOAD:TRACe<x> (No Query Form)

Load the specified saved trace from a file.

Conditions

Measurement view: Transmission Gain Where <x> is the trace 1, 2, or 3.

Group

Mass Memory commands

Syntax

```
MMEMory:TXGain:LOAD:TRACe<x> <file_name>
```

Arguments

`<file_name>::<string>` specifies the file from which to load trace data.

Examples

`MMEMory:TXGain:LOAD:TRACe1 "sample1"` loads the trace data from the sample1.tgt file.

MMEMory:TXGain:SHOW:TRACe<x>

Displays a saved trace instead of a live trace.

Conditions

Measurement view: Transmission Gain Where <x> is the trace 1, 2, or 3.

Group

Mass Memory commands

Syntax

```
MMEMory:TXGain:SHOW:TRACe<x> { OFF | ON | 0 | 1 }
MMEMory:TXGain:SHOW:TRACe<x>?
```

Arguments

`OFF` or `0` disables display of the recalled trace. ON or 1 enables display of the recalled trace.

Returns

OFF or 0 means that display of the recalled trace is disabled. ON or 1 means that display of the recalled trace is enabled.

Examples

`MMEMory:TXGain:SHOW:TRACe2 ON` enables display of the recalled data loaded in Trace 2.

MMEMory:TXGain:STORe:TRACe<x> (No Query Form)

Saves the selected trace to a file for later recall and analysis.

Conditions

Measurement view: Transmission Gain Where <x> is the trace 1, 2, or 3.

Group

Mass Memory commands

Syntax

`MMEMory:TXGain:STORe:TRACe<x> <file_name>`

Arguments

`<file_name>::<string>` specifies the file in which to store trace data. The file extension is .tgt. You can omit the extension.

Examples

`MMEMory:TXGain:STORE:TRACe1 "Sample1"` stores Trace 1 data in the Sample1.tgt file.

MMEMory:WLAN:AD:SUMMary:LIMIt:TABLe:LOAD (No Query Form)

Loads the Summary Limit table containing the limits for enabled ranges from the specified file in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Mass Memory commands

Syntax

`MMEMory:WLAN:AD:SUMMary:LIMIt:TABLe:LOAD <file name>`

Arguments

`<file name>= <String>` specifies the file to load the Summary Limit table from the file extension as *.csv*

Returns

None

Examples

`MMEMORY:WLAN:AD:SUMMARY:LIMIT:TABLE:LOAD "Table1"` loads the Summary Limit table from the Table1 *.csv* file.

MMEMory:WLAN:AD:SUMMary:LIMIt:TABLe:SAVE (No Query Form)

Stores the Summary Limit table containing the limits for enabled ranges in a specified file in the cvs format, allowing you to export the file into Microsoft Excel or other database systems in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Mass Memory commands

Syntax

`MMEMory:WLAN:AD:SUMMary:LIMIt:TABLe:SAVE <file_name>`

Arguments

`<file_name>=<string>` specifies the file to store the SEM table in the file extension as .csv.

Returns

None

Examples

`MMEMORY:WLAN:AD:SUMMARY:LIMIT:TABLE:SAVE "Table1"` stores the SEM table in the *Table1.csv file.*

MMEMory:WLAN:AY:SUMMary:LIMIt:TABLe:LOAD (No Query Form)

Loads the Summary Limit Table containing limits for enabled ranges from the specified *.csv* file.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Mass memory commands

Syntax

`MMEMory:WLAN:AY:SUMMary:LIMIt:TABLe:LOAD <file_name>`

Arguments

`<file_name>::=<string>` specifies the file to load the Summary Limit Table from a *.csv* file.

NOTE: You can omit the file extension.

Examples

`MMEM:WLAN:AY:SUMM:LIMI:TABL:LOAD "Table1"` loads the Summary Limit Table from file *Table1.csv*.

MMEMory:WLAN:AY:SUMMary:LIMIt:TABLe:SAVE (No Query Form)

Stores the Summary Limit Table containing limits for enabled ranges in a specified *.csv* file, allowing you to export the file into Microsoft Excel or other database systems.

This command will save both enabled and disabled entities.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Mass memory commands

Syntax

```
MMEMory:WLAN:AY:SUMMary:LIMIt:TABLe:SAVE <file_name>
```

Arguments

`<file_name>::=<string>` specifies the *.csv* file to store the SEM table.

NOTE: You can omit the file extension.

Examples

`MMEM:WLAN:AY:SUMM:LIMI:TABL:SAVE "Table2"` stores the SEM table in the file *Table2.csv*.

MMEMory:WLAN:CRESponse:LOAD:TRACe<x> (No Query Form)

Loads the selected WLAN Channel Response trace data from the specified file.

<x> is the selected trace, 1 or 2. 1 is the magnitude trace. 2 is the phase trace.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Mass Memory commands

Syntax

```
MMEMory:WLAN:CRESponse:LOAD:TRACe<x> <QString>
```

Arguments

Quoted string that specifies the file from which to load trace data.

Examples

`MMEMORY:WLAN:CRESPONSE:LOAD:TRACE2 "Sample1"` loads the WLAN Channel Response trace data from the Sample1 file into Trace 2 (phase trace).

MMEMory:WLAN:CRESponse:SHOW:TRACe<x>

Enables the display of a recalled WLAN Channel Response trace.

<x> is the selected trace, 1 or 2. 1 is the magnitude trace. 2 is the phase trace.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Mass Memory commands

Syntax

```
MMEMory:WLAN:CRESponse:SHOW:TRACe<x> {0|1|OFF|ON}
MMEMory:WLAN:CRESponse:SHOW:TRACe<x>?
```

Arguments

ON or 1 turns on the display of a recalled trace.

OFF or 0 turns off the display of a recalled trace.

Examples

MMEMORY:WLAN:CRESPONSE:SHOW:TRACE2 ON turns on the display of recalled trace 2.

MMEMory:WLAN:CRESponse:STORe:TRACe<x> (No Query Form)

Stores the WLAN Channel Response measurement trace data in the specified file.

<x> is the selected trace, 1 or 2. 1 is the magnitude trace. 2 is the phase trace.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Mass Memory commands

Syntax

MMEMory:WLAN:CRESponse:STORe:TRACe<x> <QString>

Arguments

Quoted string that specifies the file in which to store the trace data.

Examples

MMEMORY:OFDM:CRESPONSE:STORE:TRACE2 "Sample1" stores the trace data from Trace 2 into the Sample1 file.

TEST

Resets the trigger system and places all trigger sequences in the idle state. Any actions related to the trigger system that are in progress, such as a sweep or acquiring a measurement is also aborted.

To start data acquisition, use the INITiate commands.

Conditions

Measurement views: All

Group

Abort commands

Syntax

TEST

Related Commands

INITiate:CONTinuous (on page 1480)

INITiate:RESume (on page 1481)

Arguments

None

Examples

TEST resets the trigger system and stops data acquisition.

*OPC

Generates the operation complete message in the Standard Event Status Register (SESR) when all pending operations finish. The *OPC? query places the **ASCII** character "1" into the output queue when all pending operations are finished. The *OPC? response is not available to read until all pending operations finish.

The *OPC command allows you to synchronize the operation of the analyzer with your application program. Refer to Synchronizing execution (on page 2748) for the details.

Conditions

Measurement views: All

Group

IEEE common commands

Syntax

```
*OPC
*OPC?
```

Arguments

None

*OPT? (Query Only)

Returns a list of options installed in your analyzer.

Conditions

Measurement views: All

Group

IEEE common commands

Syntax

```
*OPT?
```

Arguments

None

Returns

The numbers of all the options installed in the analyzer, separated by commas. If no options have been installed, 0 is returned. The following table lists the options for the RSA Series analyzers.

Instrument options

Option	Description
09	Enhanced real time
10	Audio analysis

Option	Description
11	Phase noise and jitter measurement
12	Settling time measurement (frequency and phase)
21	General purpose modulation analysis
22	Flexible OFDM analysis
23	WLAN 802.11a/b/g/j/p analysis
24	WLAN 802.11n analysis
25	WLAN 802.11ac analysis
B25	25 MHz real-time capture bandwidth
B40	40 MHz real-time capture bandwidth
B85	85 MHz real-time capture bandwidth
B16x	165 MHz real-time capture bandwidth
50	1 MHz to 3/6.2 GHz pre-amplifier (RSA5103B/RSA5106B only)
51	1 MHz to 315/26.5 GHz pre-amplifier (RSA5115B/RSA5126B only)
53	Memory 4 GB acquisition memory total
56	Removable hard disk drive (not compatible with Option 59)
59	Internal hard disk drive, incompatible with Option 56
65	Digital I and Q outputs
66	Zero-span output
6566	Digital I and Q outputs and Zero-span output
PFR	Precision frequency reference
PFR50	Precision frequency reference and internal preamp (RSA5103B/RSA5126B only)
SSD	Includes additional solid-state drive for units equipped with Option 56. Minimum disk capacity is 480 GB. The Microsoft Windows 7 operating system and instrument software is preinstalled.

Examples

`*OPT?` might return 10,21,22, indicating that Option 10, 21, and 22 are currently installed in the analyzer.

Output commands

OUTPut:IF:{BANDwidth|BWIDth}

Sets or queries the IF output filter when `OUTPut:IF[:STATe]` is set to ON.

Conditions

Measurement views: All (Option 05 only)

Group

Output commands

Syntax

```
OUTPut:IF:{BANDwidth|BWIDth} { FLATtop | GAUSsian }
OUTPut:IF:{BANDwidth|BWIDth}?
```

Arguments

`FLATtop` selects the flattop filter.

`GAUSsian` selects the Gaussian filter.

Examples

`OUTPUT:IF:BANDWIDTH GAUSSIAN` selects the Gaussian filter for the IF output.

OUTPut:IF[:STATe]

Determines whether to turn on or off the 500 MHz IF Out on the rear panel.

Conditions

Measurement views: All (Option 05 only)

Group

Output commands

Syntax

```
OUTPut:IF[:STATe] { OFF | ON | 0 | 1 }
OUTPut:IF[:STATe]?
```

Related Commands

OUTPut:IF:{BANDwidth|BWIDth} (on page 1546)

Arguments

`OFF` or 0 turns off IF Out.

`ON` or 1 turns on IF Out.

Examples

`OUTPUT:IF:STATE ON` turns on IF Out.

OUTPut:IQ[:STATe]

Determines whether to enable or disable the digital IQ output data stream from the rear panel connectors.

Conditions

Measurement views: All (Option 05 only)

Group

Output commands

Syntax

```
OUTPut:IQ[:STATe] { 0 | 1 | OFF | ON }
OUTPut:IQ[:STATe]?
```

Arguments

OFF or 0 disables the digital IQ output.

ON or 1 enables the digital IQ output.

At *RST, this value is set to OFF.

Examples

OUTP:IQ:STAT ON enables the digital IQ output.

OUTP:IQ:STAT? may return OFF or 0, which means the digital IQ output is disabled.

OUTPut:NOISe[:STATe]

Determines whether to turn on or off the +28 V DC Out on the rear panel.

Conditions

Measurement views: All

Group

Output commands

Syntax

```
OUTPut:NOISe[:STATe] { OFF | ON | 0 | 1 }
OUTPut:NOISe[:STATe]?
```

Arguments

OFF or 0 turns off +28 V DC Out.

ON or 1 turns on +28 V DC Out.

Examples

OUTPUT:NOISE:STATE ON turns on +28 V DC Out.

OUTPut:IQ:SOURce

ets or queries the IQ data transfer state via the LVDS card in the CTRL7100A with option 05.

Conditions

Measurement views: All (Option 05 only for RSA7100A)

Group

Output commands

Syntax

```
OUTPut:IQ:SOURce { OFF | NORMal | CTP }
OUTPut:IQ:SOURce?
```

Arguments

`OFF` sets the IQ data transfer via LVDS to Off. This means that no streaming of data is occurring.

`NORMal` sets the IQ data transfer via LVDS to On. This means that streaming of data, if available, is occurring.

`CTP` sets the IQ data transfer via LVDS to Test (Count Pattern). This means the system is validating that the LVDS system connections are correct by sending a known counting pattern from the CTRL7100A LVDS card to the receiving LVDS card. Check the counting pattern on the receiving end to verify valid data transfer.

Examples

`OUTPUT:IQ:SOURCE?` might return NORMAL, indicating that IQ data transfer via LVDS is set to On and occurring if data is available.

OUTPut:TGen:BRIDge:MODE

Sets or queries the tracking generator bridge mode.

Conditions

Requires Option 04 in a RSA500A series or RSA600A series instrument.

Group

Output commands

Syntax

```
OUTPut:TGen:BRIDge:MODE { REFerence | CROSSover | REFLection| DISabled }
OUTPut:TGen:BRIDge:MODE
```

Arguments

Returns

Examples

```
OUTPUT:TGEN:BRIDGE:MODE
```

OUTPut:TGen:ENABle

Sets or queries the power state of the tracking generator.

Conditions

Requires Option 04 in a RSA500A series or RSA600A series instrument.

Group

Output commands

Syntax

```
OUTPut:TGen:ENABle { OFF | ON | 0 | 1 }
OUTPut:TGen:ENABle?
```

Arguments

Returns

Examples

```
OUTPUT:TGEN:ENABLE
```

OUTPut:TGen:FREQuency

Sets or queries the tracking generator frequency in Hz. The spectrum analyzer center frequency will also be set to match (track) the tracking generator frequency.

Conditions

Requires Option 04 in a RSA500A series or RSA600A series instrument.

Group

Output commands

Syntax

```
OUTPut:TGen:FREQuency <value>
OUTPut:TGen:FREQuency?
```

Arguments

Returns

Examples

```
OUTPUT:TGEN:FREQUENCY
```

OUTPut:TGen:INSTalled? (Query Only)

Queries for the presence of the tracking generator hardware.

Conditions

Requires Option 04 in a RSA500A series or RSA600A series instrument.

Group

Output commands

Syntax

```
OUTPut:TGen:INSTalled?
```

Returns

Examples

```
OUTPUT:TGEN:INSTALLED?
```

OUTPut:TGen:LEVel

Sets or queries the tracking generator output level in dBm.

Conditions

Requires Option 04 in a RSA500A series or RSA600A series instrument.

Group

Output commands

Syntax

```
OUTPut:TGen:LEVel <value>
OUTPut:TGen:LEVel?
```

Related Commands

Arguments

Returns

Examples

```
OUTPUT:TGEN:LEVEL
```

OUTPut:ZS:{BANDwidth|BWIDth}

Sets or queries the resolution bandwidth RBW filter for the Zero Span output on the rear panel connector.

Conditions

Measurement views: DPX, Spectrum

Group

Output commands

Syntax

```
OUTPut:ZS:{BANDwidth|BWIDth} <value>
OUTPut:ZS:{BANDwidth|BWIDth}?
```

Arguments

`<value>::<NRf>` specifies the resolution bandwidth.

OUTPut:ZS:{BANDwidth|BWIDth}:AUTO

Determines whether to set the Zero Span resolution bandwidth manually or automatically.

Conditions

Measurement views: DPX, Spectrum

Group

Output commands

Syntax

```
OUTPut:ZS:{BANDwidth|BWIDth}:AUTO { OFF | ON | 0 | 1 }
```

```
OUTPut:ZS:{BANDwidth|BWIDth}:AUTO?
```

Arguments

`OFF` or 0 manually sets the Zero Span output.

`ON` or 1 automatically sets the Zero Span output.

Examples

`OUTPUT:ZS:BANDWIDTH:AUTO ON` specifies that the Zero Span output will be set automatically.

OUTPut:ZS:{BANDwidth|BWIDth}:STATe

Sets or queries the state of the resolution bandwidth filter for the output connector on the rear panel.

Conditions

Measurement views: DPX, Spectrum

Group

Output commands

Syntax

```
OUTPut:ZS:{BANDwidth|BWIDth}:STATe { Auto | Manual | None }
OUTPut:ZS:{BANDwidth|BWIDth}:STATe?
```

Arguments

`Auto` sets the resolution bandwidth filter to automatic mode.

`Manual` sets the resolution bandwidth filter to manual mode.

`None` bypasses the resolution bandwidth filter.

OUTPut:ZS:DELay? (Query Only)

Queries the delay from the input connector to the Zero Span output. The delay includes the propagation time through the entire system including filters and digital processing.

Conditions

Measurement views: DPX, Spectrum

Group

Output commands

Syntax

```
OUTPut:ZS:DELay?
```

Arguments

None

Returns

`<NRf>` The actual delay of the signal through the system.

OUTPut:ZS:FILTer:STATe

Sets or queries the signal path of the filter to the Zero Span output.

Conditions

Measurement views: DPX, Spectrum

Group

Output commands

Syntax

```
OUTPut:ZS:FILTer:STATe { AUTO | HIGH | LOW }
OUTPut:ZS:FILTer:STATe?
```

Arguments

`AUTO` The filter path is calculated internally.

`HIGH` Uses the high-band 10 MHz filter.

`LOW` Uses the low-band 100 kHz filter.

OUTPut:ZS[:STATe]

Determines whether to enable or disable the Zero Span output from the rear panel connector.

Conditions

Measurement views: DPX, Spectrum

Group

Output commands

Syntax

```
OUTPut:ZS[:STATe] { OFF | ON | 0 | 1 }
OUTPut:ZS[:STATe]?
```

Arguments

`OFF` or 0 disables the Zero Span output

`ON` or 1 enables the Zero Span output

Examples

`OUTPUT:ZS:STATE ON` enables on the Zero Span output.

OUTPut:ZS:VOLTage

Sets or queries the voltage level at the Zero Span output on the rear panel connector.

Conditions

Measurement views: DPX, Spectrum

Group

Output commands

Syntax

```
OUTPut:ZS:VOLTage <value>
OUTPut:ZS:VOLTage?
```

Arguments

`<value>::<NRf>` voltage level at the Zero Span output connector.

OUTPut:ZS:VOLTage:AUTO

Sets or queries or queries how the output voltage is referenced.

Conditions

Measurement views: DPX, Spectrum

Group

Output commands

Syntax

```
OUTPut:ZS:VOLTage:AUTO { OFF | ON | 0 | 1 }
OUTPut:ZS:VOLTage:AUTO?
```

Arguments

`ON` or 1 sets the output voltage referenced to 1.0 V.

`OFF` or 0 allows users to scale down the output voltage from 0.5 V to 1.0 V.

Read commands

READ:ACPower? (Query Only)

Returns the Channel power and ACPR measurement results for all available channels.

Conditions

Measurement views: Channel power and ACPR

Group

Read commands

Syntax

```
READ:ACPower?
```

Arguments

None

Returns

`<chan_power>,<acpr_lower(1)>,<acpr_upper(1)>,`

`<acpr_lower(2)>,<acpr_upper(2)>,...`

`<acpr_lower(n)>,<acpr_upper(n)>`

Where

<chan_power> is the average power of the main channel as the power reference in dBm. The unit can be changed by the [SENSe]:POWer:UNITs command.

<acpr_lower(n)> is the ACPR for the lower channel #n in dB.

<acpr_upper(n)> is the ACPR for the upper channel #n in dB.

The number of n depends on the setting of the [SENSe]:ACPower:CHANnel:PAIRs command.

Examples

READ:ACPOWER? might return 4.227,-28.420,-23.847,-22.316,-29.225, indicating

(average power of the main channel) = 4.227 dBm,

(ACPR for the lower channel 1) = -28.420 dB,

(ACPR for the upper channel 1) = -23.847 dB,

(ACPR for the lower channel 2) = -22.316 dB, and

(ACPR for the upper channel 2) = -29.225 dB.

READ:ACPower:CHANnel:POWer? (Query Only)

Returns the average power of the main channel (power reference) in the Channel power and ACPR measurement.

Conditions

Measurement views: Channel power and ACPR

Group

Read commands

Syntax

READ:ACPower:CHANnel:POWer?

Arguments

None

Returns

<chan_power>::<NRf> is the average power of the main channel in dBm.

The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples

READ:ACPOWER:CHANNEL:POWER? might return 4.227, indicating that the average power of the main channel is 4.227 dBm.

READ:ACPower:SPECtrum? (Query Only)

Returns spectrum trace data of the Channel power and ACPR measurement.

Conditions

Measurement views: Channel power and ACPR

Group

Read commands

Syntax

`READ:ACPower:SPECtrum?`

Arguments

None

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the spectrum trace data in dBm for the point n, 4-byte little endian floating-point format specified in IEEE 488.2.

The unit can be changed by the `[SENSe]:POWer:UNITs` command.

Examples

`READ:ACPOWER:SPECTRUM?` might return #43204xxxx... (3204-byte data) for the spectrum trace data of the Channel power and ACPR measurement.

READ:{AM|FM|PM}? (Query Only)

Returns the trace data in the AM/FM/PM measurement.

Conditions

Measurement views: AM, FM, PM

Group

Read commands

Syntax

`READ:{AM|FM|PM}?`

Arguments

None

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the modulation factor in percent (AM), frequency deviation in Hz (FM), or phase deviation in degrees (PM) at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`READ:AM` might return #3156xxxx... (156-byte data) for the AM measurement trace.

READ:AM:AMINdex? (Query Only)

Returns the modulation index which is (positive peak modulation factor - negative peak modulation factor)/2, returned as a percentage (%).

Conditions

Measurement views: AM

Group

Read commands

Syntax

`READ:AM:AMINdex?`

Arguments

None

Returns

`<value>::<NRf>` the modulation index.

Examples

`READ:AM:AMINDEX?` might return 36.48, indicating the modulation index is 36.48%.

READ:AM:AMNegative? (Query Only)

Returns the negative peak modulation factor (−AM) in the AM measurement.

Conditions

Measurement views: AM

Group

Read commands

Syntax

`READ:AM:AMNegative?`

Arguments

None

Returns

`<-AM>::<NRf>` is the negative peak modulation factor in percent (%).

Examples

`READ:AM:AMNEGATIVE?` might return `-23.4`, indicating the negative peak modulation factor is −23.4%.

READ:AM:AMPositive? (Query Only)

Returns the positive peak modulation factor (+AM) in the AM measurement.

Conditions

Measurement views: AM

Group

Read commands

Syntax

`READ:AM:AMPositive?`

Arguments

None

Returns

`<+AM>::<NRf>` is the positive peak modulation factor in percent (%).

Examples

`READ:AM:AMPOSITIVE?` might return 43.8, indicating the positive peak modulation factor is 43.8%.

READ:AM:RESult? (Query Only)

Returns the AM measurement results.

Conditions

Measurement views: AM

Group

Read commands

Syntax

`READ:AM:RESult?`

Arguments

None

Returns

`<+AM>,<-AM>,<Total AM>`

Where

`<+AM>::<NRf>` is the positive peak modulation factor in percent (%).

`<-AM>::<NRf>` is the negative peak modulation factor in percent (%).

`<Total AM>::<NRf>` is the (peak to peak modulation factor)/2 in percent (%).

Examples

`READ:AM:RESULT?` might return `62.63,-50.89,56.76`.

READ:AUDio:FERRor? (Query Only)

Returns the carrier frequency error in the audio measurement. This command is only available for FM and PM measurements.

Conditions

Measurement views: Audio Summary

Group

Read commands

Syntax

`READ:AUDio:FERRor?`

Arguments

None

Returns

`<value> ::= <NRf>` the carrier frequency error.

Examples

`READ:AUDIO:FERROR?` might return 419.9529809622 indicating that the audio frequency error is approximately 420 Hz.

READ:AUDio:FREQuency? (Query Only)

Returns the audio frequency.

Conditions

Measurement views: Audio Summary

Group

Read commands

Syntax

`READ:AUDio:FREQuency?`

Arguments

None

Returns

`<value> ::= <NRf>` the audio frequency.

Examples

`READ:AUDIO:FREQUENCY?` might return 5.8239462705E+3 indicating that the audio frequency is 5.824 kHz.

READ:AUDio:HARMonic:COUNt? (Query Only)

Returns the number of harmonics in the audio spectrum measurement. The number of harmonics appear in the results table in the audio spectrum view.

Conditions

Measurement views: Audio Spectrum

Group

Read commands

Syntax

```
READ:AUDio:HARMonic:COUNt?
```

Related Commands

READ:AUDio:NHARmonic:COUNt? (on page 1562)

Arguments

None

Returns

<count> ::= <NR1> the number of harmonics in the measurement view.

Examples

READ:AUDIO:HARMONIC:COUNT? might return 3 indicating that there are three harmonics in the measurement.

READ:AUDio:HARMonic<x>:AMPLitude? (Query Only)

Returns the amplitude of the specified harmonic in the audio spectrum measurement.

The value <x> represents the specific harmonic, from 1 to the value specified by the **CALCulate:AUDio:HARMonic:HNUMber** command.

Conditions

Measurement views: Audio Spectrum

Group

Read commands

Syntax

```
READ:AUDio:HARMonic<x>:AMPLitude?
```

Related Commands

CALCulate:AUDio:HARMonic:HNUMber (on page 351)

Arguments

None

Returns

<value> ::= <NRf> the amplitude of the specified harmonic in dBc.

Examples

READ:AUDio:HARMonic3:AMPLitude? might return -2.861 indicating that amplitude of the third harmonic is -2.86 dBc.

READ:AUDio:HARMonic<x>:FREQuency? (Query Only)

Returns the frequency of the specified harmonic in the audio spectrum measurement.

The value <x> represents the specific harmonic, from 1 to the value specified by the CALCulate:AUDio:HARMonic:HNUMber command.

Conditions

Measurement views: Audio Spectrum

Group

Read commands

Syntax

`READ:AUDio:HARMonic<x>:FREQuency?`

Related Commands

CALCulate:AUDio:HARMonic:HNUMber (on page 351)

Arguments

None

Returns

`<value> ::= <NRF>` the frequency of the specified harmonic.

Examples

`READ:AUDio:HARMonic3:FREQuency?` might return 17.4718007813E+3 indicating that the frequency of the third harmonic is 17.472 kHz.

READ:AUDio:HNOise? (Query Only)

Returns the difference between the current RMS modulation value and the reference value.

This information is only available when the Hum & Noise is measurement is enabled; see the `[SENSe]:AUDio:HNOise:ENABle` command. The reference value is stored after capturing the reference by pressing the Capture Reference button or by enabling the `[SENSe]:AUDio:HNOise:REFerence` command.

Conditions

Measurement views: Audio Summary

Group

Read commands

Syntax

`READ:AUDio:HNOise?`

Related Commands

[SENSe]:AUDio:HNOise:ENABle (on page 1973)

[SENSe]:AUDio:HNOise:REFerence (on page 1974)

Arguments

None

Returns

`<value> ::= <NRF>` specifies the difference between the Hum & Noise reference and the actual signal in dB.

Examples

READ:AUDIO:HNOISE? might return 0.00 dB indicating that there is no difference between the Hum & Noise reference and the actual signal.

READ:AUDio:HNREFerence? (Query Only)

Returns the Hum and Noise RMS modulation reference value.

The value is stored after capturing the reference by pressing the Capture Reference button or by enabling the [SENSe]:AUDio:HNOise:REFerence command. This information is only available when the Hum & Noise measurement is enabled; see the [SENSe]:AUDio:HNOise:ENABle command.

For AM signal types, the reference represents modulation depth. For FM signal types, the reference represents frequency deviation. For PM signal types the reference represents phase. For Direct signal types the reference represents power.

Conditions

Measurement views: Audio Summary

Group

Read commands

Syntax

READ:AUDio:HNREFerence?

Related Commands

[SENSe]:AUDio:HNOise:ENABle (on page 1973)

[SENSe]:AUDio:HNOise:REFerence (on page 1974)

Arguments

None

Returns

<value> ::= <NRf> the Hum & Noise reference frequency.

Examples

READ:AUDIO:HNREFERENCE? might return 20.581121E+3 indicating that the Hum & Noise reference frequency is 20.58 kHz.

READ:AUDio:HPTPeak? (Query Only)

Returns the half peak-to-peak modulation excursion (where the modulation excursion depends on the signal type).

For AM signal types, the modulation excursion is “% Modulation Depth.” For FM signal types, the modulation excursion is “Frequency Deviation.” For PM signal types, the modulation excursion is “Phase Deviation.” For Direct signal types, there is no modulation excursion; it is actually “signal excursion.”

Conditions

Measurement views: Audio Summary

Group

Read commands

Syntax

READ:AUDio:HPTPeak?

Arguments

None

Returns

`<value> ::= <NRF>` specifies the half peak modulation excursion.

Examples

`READ:AUDIO:HPTPEAK?` might return 125.28231E+3 indicating that the half peak modulation excursion is 125.28 kHz.

READ:AUDio:MODDist? (Query Only)

Returns the modulation distortion for the audio measurement.

Conditions

Measurement views: Audio Summary

Group

Read commands

Syntax

`READ:AUDio:MODDist?`

Arguments

None

Returns

`<value> ::= <NRf>` specifies the modulation distortion.

Examples

`READ:AUDIO:MODDIST?` might return 98.6282113 indicating that the modulation distortion is 98.628%.

READ:AUDio:NHARmonic:COUNt? (Query Only)

Returns the number of non-harmonics in the audio spectrum measurement.

The number of non-harmonics appear in the results table in the audio spectrum view.

Conditions

Measurement views: Audio Spectrum

Group

Read commands

Syntax

`READ:AUDio:NHARmonic:COUNt?`

Related Commands

READ:AUDio:HARMonic:COUNt? (on page 1558)

Arguments

None

Returns

`<count> ::= <NR1>` the number of non-harmonics in the measurement view.

Examples

`READ:AUDIO:NHARMONIC:COUNT?` might return 7 indicating that there are seven harmonics in the measurement.

READ:AUDio:NHARmonic<x>:AMPLitude? (Query Only)

Returns the amplitude of the specified non-harmonic in the audio spectrum measurement.

The value <x> represents the specific non-harmonic, from 1 to the value specified by the `CALCulate:AUDio:HARMonic:NHNumber` command.

Conditions

Measurement views: Audio Spectrum

Group

Read commands

Syntax

`READ:AUDio:NHARmonic<x>:AMPLitude?`

Related Commands

CALCulate:AUDio:HARMonic:NHNumber (on page 352)

Arguments

None

Returns

`<value> ::= <NRf>` the amplitude of the specified non-harmonic in dBc.

Examples

`READ:AUDio:NHARMonic3:AMPLitude?` might return -2.861 indicating that amplitude of the third non-harmonic is -2.86 dBc.

READ:AUDio:NHARmonic<x>:FREQuency? (Query Only)

Returns the frequency of the specified non-harmonic in the audio spectrum measurement.

The value <x> represents the specific non-harmonic, from 1 to the value specified by the **CALCulate:AUDio:HARMonic:NHNumber** command.

Conditions

Measurement views: Audio Spectrum

Group

Read commands

Syntax

READ:AUDio:NHARmonic<x>:FREQuency?

Related Commands

CALCulate:AUDio:HARMonic:NHNumber (on page 352)

Arguments

None

Returns

<value> ::= <NRF> the frequency of the specified non-harmonic.

Examples

READ:AUDio:NHARMonic3:FREQuency? might return 17.4718007813E+3 indicating that the frequency of the third non-harmonic is 17.472 kHz.

READ:AUDio:PNEGative? (Query Only)

Returns the minus peak modulation excursion (where the modulation excursion depends on the signal type).

For AM signal types, the modulation excursion is “% Modulation Depth.” For FM signal types, the modulation excursion is “Frequency Deviation.” For PM signal types, the modulation excursion is “Phase Deviation.” For Direct signal types, there is no modulation excursion; it is actually “signal excursion.”

Conditions

Measurement views: Audio Summary

Group

Read commands

Syntax

READ:AUDio:PNEGative?

Arguments

None

Returns

Examples

READ:AUDIO:PNEGATIVE? might return -196.04321E+3 indicating that the minus peak modulation excursion is -196.04 kHz.

READ:AUDio:POWer? (Query Only)

Returns the carrier power or signal power depending the signal type.

If the signal type is Direct, the returned value is the average power of the input signal. If the signal type is AM, FM, or PM, the returned value is the carrier power; the average power of the carrier signal with the modulation removed.

Conditions

Measurement views: Audio Summary

Group

Read commands

Syntax

`READ:AUDio:POWer?`

Arguments

None

Returns

`<value> ::= <NRf>` specifies the carrier or signal power in dBm.

Examples

`READ:AUDIO:POWER?` might return -22.231123 indicating that the carrier power is -22.23 dBm.

READ:AUDio:PPOSitive? (Query Only)

Returns the positive peak modulation excursion (where the modulation excursion depends on the signal type).

For AM signal types, the modulation excursion is “% Modulation Depth.” For FM signal types, the modulation excursion is “Frequency Deviation.” For PM signal types, the modulation excursion is “Phase Deviation.” For Direct signal types, there is no modulation excursion; it is actually “signal excursion.”

Conditions

Measurement views: Audio Summary

Group

Read commands

Syntax

`READ:AUDio:PPOSitive?`

Arguments

None

Returns

`<value> ::= <NRf>` specifies the positive peak modulation excursion.

Examples

`READ:AUDIO:PPOSITIVE?` might return 215.04321E+3 indicating that the positive peak modulation excursion is -215.04 kHz.

READ:AUDio:RMS? (Query Only)

Returns the RMS modulation excursion for the audio measurement (where the modulation excursion depends on the signal type).

For AM signal types, the modulation excursion is “% Modulation Depth.” For FM signal types, the modulation excursion is “Frequency Deviation.” For PM signal types, the modulation excursion is “Phase Deviation.” For Direct signal types, there is no modulation excursion; it is actually “signal excursion.”

Conditions

Measurement views: Audio Summary

Group

Read commands

Syntax

`READ:AUDio:RMS?`

Arguments

None

Returns

`<value> ::= <NRf>` specifies the RMS modulation excursion.

Examples

`READ:AUDIO:RMS?` might return 20.575039E+3 indicating that RMS modulation excursion is 20.575 kHz.

READ:AUDio:SINad? (Query Only)

Returns the signal-to-noise and distortion for the audio measurement.

Conditions

Measurement views: Audio Summary

Group

Read commands

Syntax

`READ:AUDio:SINad?`

Arguments

None

Returns

`<value> ::= <NRf>` specifies the RMS modulation excursion.

Examples

`READ:AUDIO:SINAD?` might return 176.229024E-3 indicating that the signal-to-noise and distortion for the audio measurement is 0.18 dB.

READ:AUDio:SNOise? (Query Only)

Returns the signal level to noise level with the harmonic distortion and non-harmonic distortion components removed.

Conditions

Measurement views: Audio Summary

Group

Read commands

Syntax

`READ:AUDio:SNOise?`

Arguments

None

Returns

<value> ::= <NRf> specifies the signal level to noise level.

Examples

READ:AUDIO:SNOISE? might return -12.8156364 indicating the signal level to noise level is -12.82 dB.

READ:AUDio:SPECtrum:TRACe<x>? (Query Only)

Returns the audio spectrum trace data for the audio measurement.

The <x> represents trace 1 for this measurement. No other traces are supported.

Conditions

Measurement views: Audio Spectrum

Group

Read commands

Syntax

READ:AUDio:SPECtrum:TRACe<x>?

Arguments

None

Returns

#<num_digit><num_byte><date(1)><data(2)> . . . <data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point in percent (%), 4-byte little endian floating point format specifed in IEEE.488.2.

Examples

READ:AUDio:SPECtrum:TRACe1? might return #516004xxxx ... (16004 byte data) for the audio measurement.

READ:AUDio:THDist? (Query Only)

Returns the percentage of the total harmonic distortion in the audio measurement.

Conditions

Measurement views: Audio Summary

Group

Read commands

Syntax

```
READ:AUDio:THDist?
```

Related Commands

READ:AUDio:THDist:DB? (on page 1568)

Arguments

None

Returns

`<value> ::= <NRf>` specifies the percentage of the total harmonic distortion.

Examples

`READ:AUDIO:THDIST?` might return `53.332921`, indicating the percentage of total harmonic distortion is 53.333%.

READ:AUDio:THDist:DB? (Query Only)

Returns the total harmonic distortion in dB in the audio measurement.

Conditions

Measurement views: Audio Summary

Group

Read commands

Syntax

```
READ:AUDio:THDist:DB?
```

Related Commands

READ:AUDio:THDist? (on page 1567)

Arguments

None

Returns

`<value> ::= <NRf>` specifies the total harmonic distortion in dB.

Examples

`READ:AUDIO:THDIST:DB?` might return `-5.46009`, indicating that the total harmonic distortion is -5.46 dB.

READ:AUDio:TNHDist? (Query Only)

Returns the percentage of the total non-harmonic distortion in the audio measurement.

Conditions

Measurement views: Audio Summary

Group

Read commands

Syntax

```
READ:AUDio:TNHDist?
```

Related Commands

READ:AUDio:TNHDist:DB? (on page 1569)

Arguments

None

Returns

<value> ::= <NRf> specifies the percentage of the total non-harmonic distortion.

Examples

READ:AUDIO:TNHDIST? might return 297.332921, indicating the percentage of total non-harmonic distortion is 297.333%.

READ:AUDio:TNHDist:DB? (Query Only)

Returns the total non-harmonic distortion in dB in the audio measurement.

Conditions

Measurement views: Audio Summary

Group

Read commands

Syntax

```
READ:AUDio:TNHDist:DB?
```

Related Commands

READ:AUDio:TNHDist? (on page 1568)

Arguments

None

Returns

<value> ::= <NRf> specifies the total non-harmonic distortion in dB.

Examples

READ:AUDIO:TNHDIST:DB? might return 9.46009, indicating that the total harmonic distortion is 9.46 dB.

READ:AVTime[:MEASview<y>]:AVERage? (Query Only)

Returns the RMS (root-mean-square) value for the selected trace in the Amplitude versus Time measurement. Select the trace using the TRACe<x>:AVTime[:MEASview<y>]:SELect command.

Conditions

Measurement views: Amplitude versus Time

Group

Read commands

Syntax

`READ:AVTime[:MEASview<y>]:AVERage?`

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

`<avg>::<NRf>` is the RMS amplitude in dBm.

The unit can be changed by the `[SENSe]:POWer:UNITs` command.

Examples

`READ:AVTIME:MEASVIEW1:AVERAGE?` might return `-2.53`, indicating the RMS amplitude is -2.53 dBm on Amplitude versus Time Measurement View 1.

READ:AVTime[:MEASview<y>]:{FIRSt|SECond|THIRd|FOURth}? (Query Only)

Returns the trace data in the Amplitude versus Time measurement.

The mnemonics FIRSt, SECond, THIRd, and FOURth represent Trace 1, Trace 2, Trace 3, and Math trace, respectively. The traces can be specified by the `TRACe<x>:AVTime[:MEASview<y>]` command subgroup.

Conditions

Measurement views: Amplitude versus Time

Group

Read commands

Syntax

`READ:AVTime[:MEASview<y>]:{FIRSt|SECond|THIRd|FOURth}?`

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the amplitude in dBm at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2. The unit can be changed by the `[SENSe]:POWer:UNITs` command.

Examples

`READ:AVTIME:MEASVIEW1:FIRST?` might return #3156xxxx... (156-byte data) for Trace 1 on Amplitude versus Time Measurement View 1.

READ:AVTime[:MEASview<y>]:MAXimum? (Query Only)

Returns the maximum value for the selected trace in the Amplitude versus Time measurement. Select the trace using the `TRACe<x>:AVTime[:MEASview<y>]:SELect` command.

Conditions

Measurement views: Amplitude versus Time

Group

Read commands

Syntax

```
READ:AVTime[:MEASview<y>]:MAXimum?
```

Related Commands

READ:AVTime[:MEASview<y>]:MAXLocation? (on page 1571)

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

`<max>::<NRf>` is the maximum Amplitude in dBm.

The unit can be changed by the `[SENSe]:POWer:UNITs` command.

Examples

`READ:AVTIME:MEASVIEW1:MAXIMUM?` might return `-2.84`, indicating the maximum amplitude is -2.84 dBm for Amplitude versus Time Measurement View 1.

READ:AVTime[:MEASview<y>]:MAXLocation? (Query Only)

Returns the time at which the amplitude is maximum for the selected trace in the Amplitude versus Time measurement. Select the trace using the `TRACe<x>:AVTime[:MEASview<y>]:SELect` command.

Conditions

Measurement views: Amplitude versus Time

Group

Read commands

Syntax

```
READ:AVTime[:MEASview<y>]:MAXLocation?
```

Related Commands

READ:AVTime[:MEASview<y>]:MAXimum? (on page 1574)

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

<max_time>::<NRf> is the time at the maximum in seconds.

Examples

READ:AVTIME:MEASVIEW1:MAXLOCATION? might return 25.03E-9, indicating the amplitude is maximum at 25.03 ns on Amplitude versus Time Measurement View 1.

READ:AVTime[:MEASview<y>]:MINimum? (Query Only)

Returns the minimum value for the selected trace in the Amplitude versus Time measurement. Select the trace using the **TRACe<x>:AVTime[:MEASview<y>]:SELect** command.

Conditions

Measurement views: Amplitude versus Time

Group

Read commands

Syntax

READ:AVTime[:MEASview<y>]:MINimum?

Related Commands

READ:AVTime[:MEASview<y>]:MINLocation? (on page 1571)

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

<min>::<NRf> is the minimum amplitude in dBm. The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples

READ:AVTIME:MEASVIEW1:MINIMUM? might return -57.64, indicating the minimum amplitude is -57.64 dBm on Amplitude versus Time Measurement View 1.

READ:AVTime[:MEASview<y>]:MINLocation? (Query Only)

Returns the time at which the amplitude is minimum for the selected trace in the Amplitude versus Time measurement. Select the trace using the TRACe<x>:AVTime[:MEASview<y>]:SELect command.

Conditions

Measurement views: Amplitude versus Time

Group

Read commands

Syntax

READ:AVTime[:MEASview<y>]:MINLocation?

Related Commands

READ:AVTime[:MEASview<y>]:MINimum? (on page 1572)

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

`<min_time>::<NRf>` is the time at the minimum in seconds.

Examples

`READ:AVTIME:MEASVIEW1:MINLOCATION?` might return 450.7E-9, indicating the amplitude is minimum at 450.7 ns on Amplitude versus Time Measurement View 1.

READ:AVTime[:MEASview<y>]:RESult? (Query Only)

Returns the measurement results for the selected trace in the Amplitude versus Time measurement. Select the trace using the `TRACe<x>:AVTime[:MEASview<y>]:SELect` command.

Conditions

Measurement views: Amplitude versus Time

Group

Read commands

Syntax

`READ:AVTime[:MEASview<y>]:RESult?`

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

`<max>,<max_time>,<min>,<min_time>,<rms>`

Where

`<max>::<NRf>` is the maximum amplitude in dBm.

`<max_time>::<NRf>` is the time at the maximum in seconds.

`<min>::<NRf>` is the minimum amplitude in dBm.

`<min_time>::<NRf>` is the time at the minimum in seconds.

`<rms>::<NRf>` is the RMS amplitude in dBm.

The unit of amplitude can be changed by the `[SENSe]:POWer:UNITs` command.

Examples

`READ:AVTIME:MEASVIEW1:RESULT?` might return

-2.68,48.62E-6,-82.47,22.11E-6,-8.24, indicating that the maximum amplitude is -2.68 dBm at 48.62 µs, the minimum amplitude is -82.47 dBm at 22.11 µs, and the RMS amplitude is -8.24 dBm on Amplitude versus Time Measurement View 1.

READ:AVTime[:MEASview<y>]:AVERage:{FIRSt|SECond|THIRd|FOURth}?

Returns the RMS (root-mean-square) value for the selected trace in the Amplitude versus Time measurement. Select the trace using the `TRACe<x>:AVTime[:MEASview<y>]:SELect` command.

The mnemonics FIRSt, SECond, THIRd, and FOURth represent Trace 1, Trace 2, Trace 3, and Math trace, respectively. The traces can be specified by the `TRACe<x>:AVTime[:MEASview<y>]` command subgroup.

Conditions

Measurement views: Amplitude versus Time

Group

Read commands

Syntax

```
READ:AVTime[:MEASview<y>]:AVERage:{FIRSt|SECond|THIRd|FOURth}?
```

Related commands

READ:AVTime[:MEASview<y>]:MINimum:{FIRSt|SECond|THIRd|FOURth}? (on page 1576)

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

`<avg>::<NRf>` is the RMS amplitude in dBm.

The unit can be changed by the `[SENSe]:POWer:UNITs` command.

Examples

`READ:AVTIME:MEASVIEW1:AVERAGE:FIRST?` might return -2.53, indicating the RMS amplitude is -2.53 dBm on Trace1 for Amplitude versus Time Measurement View 1.

READ:AVTime[:MEASview<y>]:MAXimum:{FIRSt|SECond|THIRd|FOURth}?

Returns the maximum value for the selected trace in the Amplitude versus Time measurement. Select the trace using the `TRACe<x>:AVTime[:MEASview<y>]:SELect` command.

The mnemonics FIRSt, SECond, THIRd, and FOURth represent Trace 1, Trace 2, Trace 3, and Math trace, respectively. The traces can be specified by the `TRACe<x>:AVTime[:MEASview<y>]` command subgroup.

Conditions

Measurement views: Amplitude versus Time

Group

Read commands

Syntax

```
READ:AVTime[:MEASview<y>]:MAXimum:{FIRSt|SECond|THIRd|FOURth}?
```

Related commands

READ:AVTime[:MEASview<y>]:MAXLocation:{FIRSt|SECond|THIRd|FOURth}?

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

<max>::<NRf> is the maximum Amplitude in dBm.

The unit can be changed by the [:SENSe]:POWer:UNITs command.

Examples

READ:AVTIME:MEASVIEW1:MAXIMUM:FIRST? might return -2.84, indicating the maximum amplitude is -2.84 dBm on Trace1 for Amplitude versus Time Measurement View 1.

READ:AVTime[:MEASview<y>]:MAXLocation:{FIRSt|SECond|THIRd|FOURth}?

Returns the time at which the amplitude is maximum for the selected trace in the Amplitude versus Time measurement. Select the trace using the TRACe<x>:AVTime[:MEASview<y>]:SELect command.

The mnemonics FIRSt, SECond, THIRd, and FOURth represent Trace 1, Trace 2, Trace 3, and Math trace, respectively. The traces can be specified by the TRACe<x>:AVTime[:MEASview<y>] command subgroup.

Conditions

Measurement views: Amplitude versus Time

Group

Read commands

Syntax

READ:AVTime[:MEASview<y>]:MAXLocation:{FIRSt|SECond|THIRd|FOURth}?

Related commands

READ:AVTime[:MEASview<y>]:MAXimum:{FIRSt|SECond|THIRd|FOURth}? (on page 1574)

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

<max>::<NRf> is the maximum Amplitude in dBm.

Examples

READ:AVTIME:MEASVIEW1:MAXLOCATION:FIRST? might return 25.03E-9, indicating the amplitude is maximum at 25.03 ns on Trace1 for Amplitude versus Time Measurement View 1.

READ:AVTime[:MEASview<y>]:MINimum:{FIRSt|SECond|THIRd|FOURth}?

Returns the minimum value for the selected trace in the Amplitude versus Time measurement. Select the trace using the `TRACe<x>:AVTime[:MEASview<y>]:SELect` command.

The mnemonics FIRSt, SECond, THIRd, and FOURth represent Trace 1, Trace 2, Trace 3, and Math trace, respectively. The traces can be specified by the `TRACe<x>:AVTime[:MEASview<y>]` command subgroup.

Conditions

Measurement views: Amplitude versus Time

Group

Read commands

Syntax

```
READ:AVTime[:MEASview<y>]:MINimum:{FIRSt|SECond|THIRd|FOURth}?
```

Related commands

READ:AVTime[:MEASview<y>]:MINLocation:{FIRSt|SECond|THIRd|FOURth}? (on page 1576)

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

`<min>::<NRf>` is the minimum amplitude in dBm. The unit can be changed by the `[SENSe]:POWer:UNITs` command.

Examples

`READ:AVTIME:MEASVIEW1:MINIMUM:FIRST?` might return `-57.64`, indicating the minimum amplitude is -57.64 dBm on Trace1 for Amplitude versus Time Measurement View 1.

READ:AVTime[:MEASview<y>]:MINLocation:{FIRSt|SECond|THIRd|FOURth}?

Returns the time at which the amplitude is minimum for the selected trace in the Amplitude versus Time measurement. Select the trace using the `TRACe<x>:AVTime[:MEASview<y>]:SELect` command.

The mnemonics FIRSt, SECond, THIRd, and FOURth represent Trace 1, Trace 2, Trace 3, and Math trace, respectively. The traces can be specified by the `TRACe<x>:AVTime[:MEASview<y>]` command subgroup.

Conditions

Measurement views: Amplitude versus Time

Group

Read commands

Syntax

```
READ:AVTime[:MEASview<y>]:MINLocation:{FIRSt|SECond|THIRd|FOURth}?
```

Related commands

READ:AVTime[:MEASview<y>]:MINimum:{FIRSt|SECond|THIRd|FOURth}? (on page 1576)

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

`<min>::<NRf>` is the minimum amplitude in dBm. The unit can be changed by the `[:SENSe]:POWer:UNITs` command.

Examples

`READ:AVTIME:MEASVIEW1:MINLOCATION:FIRST?` might return 450.7E-9, indicating the amplitude is minimum at 450.7 ns on Trace1 for Amplitude versus Time Measurement View 1.

READ:AVTime[:MEASview<y>]:OGRAm? (Query only)

Returns the ogram trace data. The line is selected using the `TRACe:AVTime:MEASview<y>:OGRAm:SELect:LINE` command.

Conditions

Applicable measurement view must be active.

Group

Read commands

Syntax

`READ:AVTime[:MEASview<y>]:OGRAm?`

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional. By default y is 1.

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in <num_byte>.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the trace data in dBm for the point n, 4-byte little endian floating-point format specified in IEEE 488.2.

The unit can be changed by the `[SENSe]:POWer:UNITs` command.

Examples

`READ:AVTime:MEASview1:OGRAm?` might return #43204xxxx... (3204-byte data) for the ogram trace.

READ:AVTime[:MEASview<y>]:RESult:{FIRSt|SECond|THIRd|FOURth}?

Returns the measurement results for the selected trace in the Amplitude versus Time measurement. Select the trace using the `TRACe<x>:AVTime[:MEASview<y>]:SELect` command.

The mnemonics FIRSt, SECond, THIRd, and FOURth represent Trace 1, Trace 2, Trace 3, and Math trace, respectively. The traces can be specified by the `TRACe<x>:AVTime[:MEASview<y>]` command subgroup.

Conditions

Measurement views: Amplitude versus Time

Group

Read commands

Syntax

```
READ:AVTime[:MEASview<y>]:RESult:{FIRSt|SECond|THIRd|FOURth}?
```

Related commands

READ:AVTime[:MEASview<y>]:MINimum:{FIRSt|SECond|THIRd|FOURth}? (on page 1576)

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

`<max>,<max_time>,<min>,<min_time>,<rms>`

Where

`<max>::<NRf>` is the maximum amplitude in dBm.

`<max_time>::<NRf>` is the time at the maximum in seconds.

`<min>::<NRf>` is the minimum amplitude in dBm.

`<min_time>::<NRf>` is the time at the minimum in seconds.

`<rms>::<NRf>` is the RMS amplitude in dBm.

The unit of amplitude can be changed by the `[SENSe]:POWer:UNITs` command.

Examples

`READ:AVTIME:MEASVIEW1:RESULT:FIRST?` might return `-2.68,48.62E-6,-82.47,22.11E-6,-8.24`, indicating that the maximum amplitude is -2.68 dBm at 48.62 µs, the minimum amplitude is -82.47 dBm at 22.11 µs, and the RMS amplitude is -8.24 dBm on Trace1 for Amplitude versus Time Measurement View 1.

READ:AVTime[:MEASview<y>]:TRACe:BITMap? (Query Only)

Acquires a waveform and then returns trace waveform data of the bitmap trace in the Amplitude versus Time measurement.

Conditions

Measurement views: Amplitude versus Time

Group

Read commands

Syntax

```
READ:AVTime[:MEASview<y>]:TRACe:BITMap?
```

Arguments

None

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Returns

The format is a binary block of 32 bit floating point values. Each value ranges from 0 to 1.0, where 1.0 represents 100%.

Examples

`READ:AVTIME:TRACE:BITMAP?` might return #42004xxxx... (2004-byte data) for the waveform data of the bitmap trace.

READ:BIBEmissions:FTX? (Query Only)

Returns the band number corresponding to the transmitted FTX detected in the Bluetooth InBand Emission display.

Conditions

Measurement views: Bluetooth InBand Emission

Group

Read commands

Syntax

`READ:BIBEmissions:FTX?`

Arguments

None

Returns

`<value> ::= <NR1>` the band number.

Examples

`READ:BIBEMISSIONS:FTX?` might return `39`, indicating the band number corresponding to transmitted FTX.

READ:BIBEmissions:POWer? (Query Only)

Returns all channel power results from the Bluetooth InBand Emission display.

Conditions

Measurement views: Bluetooth InBand Emission

Group

Read commands

Syntax

`READ:BIBEmissions:POWer?`

Arguments

None

Returns

<Band0_power>,<Band1_power>,<Band2_power>, . . . <Band78_power>

Where:

Bandx_power is the integrated power in that band. All power values are in dBm

Examples

READ:BIBEMISSIONS:POWER? might return 4.227,-28.420,-23.847, . . . ,-29.225, indicating: (integrated power for Band 0) = 4.227 dBm,

(Integrated power for Band 1) = -28.420 dBm,

(Integrated power for Band 2) = -23.847 dBm, and

(Integrated power for Band 78) = -29.225 dBm.

READ:BIBEmissions:RESUlts:STATus? (Query Only)

Returns the Pass/Fail result status of the Bluetooth InBand Emission display.

Conditions

Measurement views: Bluetooth InBand Emission

Group

Read commands

Syntax

READ:BIBEmissions:RESUlts:STATus?

Arguments

None

Examples

READ:BIBEMISSIONS:RESULTS:STATUS? returns the Pass/Fail status result on the display.

READ:BLUEtooth:CONSte:FERRor? (Query Only)

Queries the frequency error in Hz in the Bluetooth Constellation. The frequency error is the difference between the measured carrier frequency of the signal and the user-selected center frequency.

Conditions

Measurement views: Bluetooth Constellation

Group

Read commands

Syntax

READ:BLUEtooth:CONSte:FERRor?

Arguments

None

Returns

<freq_error> ::= <NRf> the frequency error.

Examples

READ:BLUEtooth:CONSte:FERRor? might return -10.7E+3, which is a frequency error of -10.7 kHz.

READ:BLUEtooth:CONSte:TRACe? (Query Only)

Queries the Bluetooth Constellation trace data.

Conditions

Measurement views: Bluetooth Constellation

Group

Read commands

Syntax

READ:BLUEtooth:CONSte:TRACe?

Arguments

None

Returns

#<num_digit><num_byte><I(1)><Q1><I(2)><Q2> . . . <I(n)><Qn>

Where:

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<I(n)> and <Qn> are the normalized I- and Q-coordinate values at the nth data point. 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

READ:BLUEtooth:CONSte:TRACe? might return #43848xxxx . . . (3848-byte data) for Bluetooth Constellation trace.

READ:BLUEtooth:EDIagram:FDEViation? (Query Only)

Returns the frequency deviation vs. time trace data with the X values in the Bluetooth eye diagram.

Conditions

Measurement views: Bluetooth eye diagram.

Group

Read commands

Syntax

READ:BLUEtooth:EDIagram:FDEViation?

Arguments

None

Returns

`#<num_digits><num_bytes><Y(1)><X(1)><Y(2)><X(2)> . . . <Y(n)><X(n)>`

Where

`<num_digits>` is the number of bytes in `<num_bytes>`.

`<num_bytes>` is the number of bytes of data that follow.

`<Y(1)>` and `<X(1)>` are the normalized I- and Q-coordinate values at the nth datapoint. The 4-byte little endian floating point format is specified in IEEE 488.2.

Examples

`READ:BLUEtooth:EDIagram:FDEVIiation?` might return #3160xxxx... (160-byte data) for the frequency deviation versus time trace.

READ:BLUEtooth:EDIagram:FERRor? (Query Only)

Returns the frequency error in the Bluetooth eye diagram measurement.

Conditions

Measurement views: Bluetooth eye diagram.

Group

Read commands

Syntax

`READ:BLUEtooth:EDIagram:FERRor?`

Arguments

None

Returns

`<freq_error> ::= <NRf>` the frequency error in Hz.

Examples

`READ:BLUEtooth:EDIagram:FERRor?` might return `-10.7E+3`, indicating the frequency error is -10.7 kHz.

READ:BLUEtooth:EDIagram:HORiz:SCALe? (Query Only)

Returns the value of the horizontal scale in the Bluetooth eye diagram.

Conditions

Measurement views: Bluetooth eye diagram.

Group

Read commands

Syntax

`READ:BLUEtooth:EDIagram:HORiz:SCALe?`

Arguments

None

Returns

<value> ::= <NR3> the horizontal scale value.

Examples

READ:BLUETOOTH:EDIAGRAM:HORIZ:SCALE? might return 2 indicating the horizontal scale value is 2 symbols.

READ:BLUEtooth:EDIagram:I? (Query Only)

Returns the I vs. time trace in the Bluetooth eye diagram.

Conditions

Measurement views: Bluetooth eye diagram.

Group

Read commands

Syntax

READ:BLUEtooth:EDIagram:I?

Arguments

None

Returns

#<num_digits><num_bytes><Y(1)><X(1)><Y(2)><X(2)> . . . <Y(n)><X(n)>

Where

<num_digits> is the number of bytes in <num_bytes>.

<num_bytes> is the number of bytes of data that follow.

<Y(1)> and <X(1)> are the normalized I- and Q-coordinate values at the nth datapoint. The 4-byte little endian floating point format is specified in IEEE 488.2.

Examples

READ:BLUEtooth:EDIagram:I? might return #3160xxxx... (160-byte data) for the I vs. time trace.

READ:BLUEtooth:EDIagram:Q? (Query Only)

Returns the Q vs. time trace in the Bluetooth eye diagram.

Conditions

Measurement views: Bluetooth eye diagram.

Group

Read commands

Syntax

READ:BLUEtooth:EDIagram:Q?

Arguments

None

Returns

#<num_digits><num_bytes><Y(1)><X(1)><Y(2)><X(2)> . . . <Y(n)><X(n)>

Where

<num_digits> is the number of bytes in <num_bytes>.

<num_bytes> is the number of bytes of data that follow.

<Y(1)> and <X(1)> are the normalized I- and Q-coordinate values at the nth datapoint. The 4-byte little endian floating point format is specified in IEEE 488.2.

Examples

READ:BLUEtooth:EDIagram:Q? might return #3160xxxx... (160-byte data) for the Q vs. time trace.

READ:BLUEtooth:FDVTime:TRACe? (Query Only)

Returns the Bluetooth Frequency Deviation vs. Time trace data.

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Read commands

Syntax

READ:BLUEtooth:FDVTime:TRACe?

Arguments

None

Returns

#<num_digits><num_bytes><data(1)><data(2)> . . . <data(n)>

Where

.<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the frequency deviation in Hz at the nth datapoint. The 4-byte little endian floating point format is specified in IEEE 488.2.

Examples

READ:BLUEtooth:FDVTime:TRACe? might return #3160xxxx... (160-byte data) for the trace.

READ:BLUEtooth:FREQuency:ERROr? (Query Only)

Returns the frequency error from the Bluetooth Constellation diagram.

Conditions

Measurement views: Bluetooth Constellation

Group

Read commands

Syntax

```
READ:BLUEtooth:FREQuency:ERROr?
```

Arguments

None

Returns

<NR3>, the frequency error value in Hz.

Examples

READ:BLUETOOTH:FREQUENCY:ERROR? might return 0.0000, indicating the frequency error is 0.0000 Hz.

READ:BLUEtooth:FREQuency:ERROr:TYPE

Returns the frequency error type in the Bluetooth Constellation display.

Conditions

Measurement views: Bluetooth Constellation

Group

Read commands

Syntax

```
READ:BLUEtooth:FREQuency:ERROr:TYPE
```

Arguments

None

Returns

1 indicates the error is automatically detected.

0 indicates the error is not automatically detected.

Examples

READ:BLUETOOTH:FREQUENCY:ERROR:TYPE might return 1, indicating the frequency error is automatically detected.

READ:BLUEtooth:FREQuency:OFFSet:DRIFt:F1FZero? (Query Only)

Returns the drift in frequency offset (f1−f0 − the frequency offset calculated in the first interval in the payload − the frequency offset calculated in the preamble) in the Bluetooth Frequency Deviation vs. Time display.

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Read commands

Syntax

`READ:BLUEtooth:FREQuency:OFFSet:DRIFt:F1FZero?`

Arguments

None

Returns

`<NR3>` the drift in frequency offset (f1−f0 − the frequency offset calculated in the first interval in the payload − the frequency offset calculated in the preamble)

Examples

`READ:BLUEtooth:FREQuency:OFFSet:DRIFt:F1FZero?` might return `−207.6465301514` indicating the drift in frequency offset (f1-f0 − the frequency offset calculated in the first interval in the payload − the frequency offset calculated in the preamble).

READ:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFN5? (Query Only)

Returns the maximum drift of the frequency offset in payload intervals spaced 50 µs away in the Bluetooth Frequency Deviation vs. Time display.

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Read commands

Syntax

`READ:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFN5?`

Arguments

None

Returns

`<NR3>` the maximum drift of the frequency offset in payload intervals spaced 50 µs away.

Examples

`READ:BLUEtooth:FREQuency:OFFSet:DRIFt:FNF5?` might return `−207.6465301514`, indicating the maximum drift of the frequency offset in payload intervals spaced 50 µs away.

READ:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFN5:INTErval? (Query Only)

Returns the interval at which the maximum drift of frequency offset in a duration of 50 µs occurred.

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Read commands

Syntax

`READ:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFN5:INTErval?`

Arguments

None

Returns

`<NR1>` interval at which the maximum drift of frequency offset in a duration of 50 µs occurred.

Examples

`READ:BLUEtooth:FREQuency:OFFSet:DRIFt:FNF5:INTErval?` might return `16`, indicating the interval at which the maximum drift of frequency offset in a duration of 50 µs occurred. Here 16 refers to the 16th interval and that the drift of frequency offset calculated in the 16th interval from that calculated in the 11th interval is the maximum among those calculated 50 µs away.

READ:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFZERO? (Query Only)

Returns the maximum drift of the frequency offset of the intervals in the payload from the preamble packet in the Bluetooth Frequency Deviation vs. Time display.

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Read commands

Syntax

`READ:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFZERO?`

Arguments

None

Returns

`<NR3>` the maximum drift of frequency offset calculated in payload intervals from the offset calculated in the preamble.

Examples

`READ:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFZero?` might return `-207.6465301514`, indicating that the maximum drift of frequency offset calculated in payload intervals from the offset calculated in the preamble is -207.6465301515 Hz.

READ:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFZERO:INTErval? (Query Only)

Returns the interval (n) at which the maximum drift fn-f0 occurred in the Bluetooth Frequency Deviation vs. Time display. (The fn-f0 indicates the drift of the frequency offset calculated in each payload interval from the offset calculated in the preamble.)

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Read commands

Syntax

```
READ:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFZERO:INTErval?
```

Arguments

None

Returns

<NR1> the interval at which the maximum drift (fn-f0) occurred.

Examples

READ:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFZero:INTErval? might return 16, indicating that at the 16th payload interval, the maximum drift occurred. f(16) - f(0) is the maximum drift among all f(n) -f (0).

READ:BLUEtooth:FREQuency:OFFSet:MAX? (Query Only)

Returns the maximum frequency offset in the Frequency Deviation vs. Time and the Center Frequency Offset and Drift displays.

Conditions

Measurement views: Frequency Deviation vs. Time, CF Offset

Group

Read commands

Syntax

```
READ:BLUEtooth:FREQuency:OFFSet:MAX?
```

Arguments

None

Returns

<NR3> the maximum frequency offset value.

Examples

READ:BLUEtooth:FREQuency:OFFSet:MAX? might return -189.4632263184, the frequency offset value.

READ:BLUEtooth:FREQuency:OFFSet:MAX:INTErval? (Query Only)

Returns the maximum frequency offset interval in the Frequency Deviation vs. Time and the Center Frequency Offset and Drift displays.

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Read commands

Syntax

`READ:BLUEtooth:FREQuency:OFFSet:MAX:INTErval?`

Arguments

None

Returns

`<NR1>` the maximum frequency offset interval.

Examples

`READ:BLUEtooth:FREQuency:OFFSet:MAX:INTErval?` might return `271`, indicating that the maximum drift f(n) - f(0) occurred at the 271st payload interval.

READ:BLUEtooth:FREQuency:OFFSet:PREAmble? (Query Only)

Returns the offset calculated in the preamble region in the Frequency Deviation vs. Time and the Center Frequency Offset and Drift displays.

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Read commands

Syntax

`READ:BLUEtooth:FREQuency:OFFSet:PREAmble?`

Arguments

None

Returns

`<NR3>` the offset calculated in the preamble region.

Examples

`READ:BLUEtooth:FREQuency:OFFSet:PREAmble?` might return `106.4204711914`, indicating the offset calculated in the preamble region.

READ:BLUEtooth:MODUlation:CHARacteristics:AVErage:F? (Query Only)

Returns the average frequency value when the selected test pattern is Other in the Bluetooth Frequency Deviation vs. Time display.

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Read commands

Syntax

```
READ:BLUEtooth:MODUlation:CHARacteristics:AVErage:F?
```

Arguments

None

Returns

<NRf> the average frequency.

Examples

READ:BLUEtooth:MODUlation:CHARacteristics:AVErage:F? might return 140.8309531250E+3, the average frequency value.

READ:BLUEtooth:MODUlation:CHARacteristics:AVErage:F2F1ratio? (Query Only)

Returns the average ratio value for the high and low frequency deviation in the Bluetooth Frequency Deviation vs. Time display.

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Read commands

Syntax

```
READ:BLUEtooth:MODUlation:CHARacteristics:AVErage:F2F1ratio?
```

Arguments

None

Returns

<NR1> the average ratio value.

Examples

READ:BLUEtooth:MODUlation:CHARacteristics:AVErage:F2F1ratio? might return ????, the average ratio value.

READ:BLUEtooth:MODUlation:CHARacteristics:AVErage:FONE? (Query Only)

Returns the average frequency deviation for the low deviation pattern in the Bluetooth Frequency Deviation vs. Time display.

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Read commands

Syntax

READ:BLUEtooth:MODUlation:CHARacteristics:AVErage:FONE?

Arguments

None

Examples

READ:BLUEtooth:MODUlation:CHARacteristics:AVErage:FONE? might return 139.8309531250E+3, the average frequency deviation for the low deviation pattern.

READ:BLUEtooth:MODUlation:CHARacteristics:AVErage:FTWO? (Query Only)

Returns the average frequency deviation for the high deviation pattern in the Bluetooth Frequency Deviation vs. Time display.

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Read commands

Syntax

READ:BLUEtooth:MODUlation:CHARacteristics:AVErage:FTWO?

Arguments

None

Examples

READ:BLUEtooth:MODUlation:CHARacteristics:AVErage:FTWO? might return 139.8309531250E+3, the average frequency deviation for the high deviation pattern.

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:COUNt? (Query Only)

Returns the number of packets analyzed to obtain average drift results from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:COUNt?

Arguments

None

Returns

<NR1> the number of packets analyzed.

Examples

`READ:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:COUNT?` might return `8`, the number of packets analyzed.

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:F1FZero? (Query Only)

Returns the drift f1−f0 from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:F1FZero?`

Arguments

None

Returns

`<NRf>` the drift f1−f0 value.

Examples

`READ:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:F1FZERO?` might return 2.846E+3 indicating that the drift f1-f0 value is 2.846 kHz.

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:F1FZero:STATus? (Query Only)

Returns the Pass/Fail drift f1−f0 result status in the packet from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:F1FZero:STATus?`

Arguments

None

Returns

`<string>` the Pass/Fail result status.

Examples

READ:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:F1FZERO:STATUS? might return PASS, FAIL, or NA indicating whether the drift f1−f0 is within the specified limit.

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFN5? (Query Only)

Returns the maximum drift of the fn−fn-5 value from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFN5?

Arguments

None

Returns

<NRf> the fn−fn-5 value.

Examples

READ:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:FNFN5? might return 2.846E+3 indicating that the maximum drift fn−fn-5 value in the preamble region is -2.846 kHz.

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFN5:STATus? (Query Only)

Returns the Pass/Fail status of the maximum fn−fn-5 drift in the packet from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFN5:STATus?

Arguments

None

Returns

<string> the Pass/Fail result status.

Examples

READ:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:FNFN5:STATUS? might return PASS, FAIL, or NA indicating whether the maximum drift fn−fn−5 is within the specified limit.

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFZero? (Query Only)

Returns the maximum drift fn−f0 from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFZero?

Arguments

None

Returns

<NRf> the fn−fn0 value.

Examples

READ:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:FNFZERO? might return 4.846E+3 indicating that the drift fn−f0 value in the preamble region is 4.846 kHz.

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFZero:STATus? (Query Only)

Returns the Pass/Fail result status of the fn−f0 maximum drift in the packet from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFZero:STATus?

Arguments

None

Returns

<string> the Pass/Fail result status.

Examples

READ:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:FNFZERO:STATUS? might return PASS, FAIL, or NA indicating whether the maximum drift fn−f0 is within the specified limit.

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded:CODIng:SCHEme (Query Only)

Returns the offset drift coding scheme value (LE Coded) in Bluetooth Summary display.

Conditions

Measurement view: Bluetooth Summary display

Group

Read commands

Syntax

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded:CODIng:SCHEme?

Arguments

None

Returns

<NR1> 2 or 8

Where 2 is data at 500 Kb/s and 8 is 125 Kb/s

Examples

READ:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:LECODED:CODING:SCHEME might return "8", indicating the offset drift coding scheme value is 8.

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded:F0FThree (Query Only)

Returns the Drift value f0-f3 for LE Coded in Bluetooth Summary display.

Conditions

Measurement view: Bluetooth Summary display

Group

Read commands

Syntax

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded:F0FThree?

Arguments

None

Returns

<NRf> Maximum Frequency value in kHz

Examples

`READ:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:LECODED:F0FTHREE` might return `207.2 kHz`, indicating the f0-f3 drift value is 207.2 kHz.

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded:F0FThree:STATus (Query Only)

Returns the Drift value Status of f0-f3 for LE Coded in Bluetooth Summary display.

Conditions

Measurement view: Bluetooth Summary display

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded:F0FThree:STATus?`

Arguments

None

Returns

`<String>` PASS , FAIL or NA

`PASS` indicates that the test passed

`FAIL` indicates that the test failed

`NA` indicates that the test is not Applicable

Examples

`READ:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:LECODED:F0FTHREE:STATUS` might return `Pass`, indicating the Drift value Status of f0-f3 was Pass.

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded:F1FFour (Query Only)

Returns the Drift value of f1-f4 for LE Coded in Bluetooth Summary display.

Conditions

Measurement view: Bluetooth Summary display

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded:F1FFour?`

Arguments

None

Returns

`<NRf>` Maximum Frequency value in kHz

Examples

`READ:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:LECODED:F1FFOUR` might return `-10.22 kHz`, indicating the f1-f4 drift value is -10.22 kHz.

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded:F1FFour:STATus (Query Only)

Returns the Drift value Status of f1-f4 for LE Coded in Bluetooth Summary display.

Conditions

Measurement view: Bluetooth Summary display

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded:F1FFour:STATus?`

Arguments

None

Returns

`<String>` PASS , FAIL or NA

`PASS` indicates that the test passed

`FAIL` indicates that the test failed

`NA` indicates that the test is not Applicable

Examples

`READ:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:LECODED:F1FFOUR:STATUS` might return `Pass`, indicating the Drift value Status of f1-f4 was Pass.

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded:FNFN3 (Query Only)

Returns the Drift value of fn-f(n-3) for LE Coded in Bluetooth Summary display.

Conditions

Measurement view: Bluetooth Summary display

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded:FNFN3?`

Arguments

None

Returns

<NRf> Maximum Frequency value in kHz

Examples

READ:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:LECODED:FNFN**3** might return –10.42 kHz, indicating the fn-f(n-3) drift value is -10.42 kHz.

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded:FNFN3:STATus (Query Only)

Returns the Drift value Status of fn-f(n-3) for LE Coded in Bluetooth Summary display.

Conditions

Measurement view: Bluetooth Summary display

Group

Read commands

Syntax

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:LECoded:FNFN3:STATus?

Arguments

None

Returns

<String> PASS , FAIL or NA

PASS indicates that the test passed

FAIL indicates that the test failed

NA indicates that the test is not Applicable

Examples

READ:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:LECODED:FNFN3:STATUS might return Pass, indicating the Drift value Status of fn-f(n-3) was Pass.

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:MAX? (Query Only)

Returns the maximum frequency offset present in the packet from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:MAX?

Arguments

None

Returns

<NRf> the maximum frequency offset value in the packet.

Examples

READ:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:MAX? might return 7.446E+3, indicating that the maximum frequency offset value in the packet is 7.446 kHz.

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:MAX:STATus? (Query Only)

Returns the Pass/Fail result status of the frequency offset maximum in the packet from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:MAX:STATus?

Arguments

None

Returns

<string> the Pass/Fail result status.

Examples

READ:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:MAX:STATUS? might return PASS, FAIL, or NA, indicating whether the maximum frequency offset in the packet is within the specified limit.

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:PREAmble? (Query Only)

Returns the frequency offset value in the preamble region from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:PREAmble?

Arguments

None

Returns

<NRf> the frequency offset value in the preamble region.

Examples

READ:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:PREAMBLE? might return 3.546E+3, indicating that the frequency offset value in the preamble region is 3.546 kHz.

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:PREAmble:STATus? (Query Only)

Returns the Pass/Fail result status of the frequency offset in the preamble region.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:PREAmble:STATus?

Arguments

None

Returns

<string> the Pass/Fail result status.

Examples

READ:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:PREAMBLE:STATUS? might return PASS, FAIL, or NA, indicating whether the frequency offset in the preamble region is within the specified limit.

READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:F:MAX:PERCentage:COUNt? (Query Only)

Returns the number of packets analyzed out of 10 packets for ΔFavg from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:F:MAX:PERCentage:COUNt?

Arguments

None

Returns

<NR1> the number of packets analyzed.

Examples

READ:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:DELTA:F:MAX:PERCENTAGE:COUNT? might return 3, indicating that three packets were analyzed.

READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:FONE:MAX:PERCentage:COUNt? (Query Only)

Returns the number of packets analyzed out of 10 packets for the ΔF1avg from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:FONE:MAX:PERCentage:COUNt?

Arguments

None

Returns

<NR1> the number of packets analyzed.

Examples

READ:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:DELTA:FONE:MAX:PERCENTAGE:COUNT? might return 6 indicating that six packets were analyzed.

READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:FTWO:MAX:PERCentage? (Query Only)

Returns the percentage of the ΔF2max% values that are above the specified limit.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:FTWO:MAX:PERCentage?

Arguments

None

Returns

<NRf> the number of packets analyzed.

Examples

READ:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:DELTA:FTWO:MAX:PERCENTAGE? might return 99.2 indicating that 99.2% of the deltaF2 maximum values are above the specified limit.

READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:FTWO:MAX:PERCentage:COUNt? (Query Only)

Returns the number of packets analyzed out of 10 packets for delta F2 average from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:FTWO:MAX:PERCentage:COUNt?

Arguments

None

Returns

<NR1> the number of packets analyzed.

Examples

READ:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:DELTA:FTWO:MAX:PERCENTAGE:COUNT? might return 6 indicating that six packets were analyzed.

READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:FTWO:MAX:PERCentage:STATus? (Query Only)

Returns the Pass/Fail result status of the ΔF2Max% field from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:FTWO:MAX:PERCentage:STATus?

Arguments

None

Returns

<string> the Pass/Fail result status.

Examples

READ:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:DELTA:FTWO:MAX:PERCENTAGE:STATUS? might return PASS, FAIL, or NA indicating the Pass/Fail status of the ΔF2Max% field in the Bluetooth summary.

READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:F2F1ratio? (Query Only)

Returns the ratio of the ΔF2avg to ΔF1avg from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:F2F1ratio?

Arguments

None

Returns

<NRf> the value of the ΔF1avg to ΔF2avg ratio.

Examples

READ:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:F2F1RATIO? might return 225.12E+3, indicating the average ratio is 22.5.12 kHz.

READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:F2F1ratio:STATus? (Query Only)

Returns the Pass/Fail results status of the ΔF2avg/ΔF1avg field from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:F2F1ratio:STATus?

Arguments

None

Returns

<string> the Pass/Fail result status.

Examples

READ:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:F2F1RATIO:STATUS? might return PASS, FAIL, or NA, indicating the Pass/Fall result status of the ΔF2avg/ΔF1avg field.

READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:F? (Query Only)

Returns the ΔFavg value from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:F?

Arguments

None

Returns

<NRf> the ΔFavg value.

Examples

READ:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:F? might return 2.8723E+3, indicating the ΔFavg is 2.8723 kHz.

READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:FONE? (Query Only)

Returns the delta F1 average value from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:FONE?

Arguments

None

Returns

<NRf> the delta F1 value.

Examples

`READ:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:FONE?` might return `155.3672E+3`, indicating the delta F1 average is 155.3672 kHz.

READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:FONE:STATus? (Query Only)

Returns the Pass/Fail results status of the delta F1 average from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:FONE:STATus?`

Arguments

None

Returns

`<string>` the Pass/Fail results status.

Examples

`READ:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:FONE:STATUS?` might return `PASS`, `FAIL`, or `NA`, indicating whether the delta F1 average is within the limit.

READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:FTWO? (Query Only)

Returns the delta F2 average value from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:FTWO?`

Arguments

None

Returns

`<NRf>` the delta F1 value.

Examples

`READ:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:FTWO?` might return `225.12E+3`, indicating that the delta F2 value is 225.12 kHz.

READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:INDex (Query Only)

Returns the Modulation index value in Bluetooth Summary display.

Conditions

Measurement view: Bluetooth Summary display

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:INDex?`

Arguments

None

Returns

`<String>`

Examples

`READ:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:INDEX` might return `"Stable"`, indicating the status of modulation index is Stable.

READ:BLUEtooth:SUMMary:OUTPut:POWEr:AVErage? (Query Only)

Returns the average output power from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:OUTPut:POWEr:AVErage?`

Arguments

None

Returns

`<NRf>` the average output power value in dBm.

Examples

`READ:BLUETOOTH:SUMMARY:OUTPUT:POWER:AVERAGE?` might return -29.1 indicating that the average output power is -29.1 dBm.

READ:BLUEtooth:SUMMary:OUTPut:POWEr:AVErage:STATus? (Query Only)

Returns the Pass/Fail results status of the average output power from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:OUTPut:POWEr:AVErage:STATus?`

Arguments

None

Returns

`<string>` the Pass/Fail results status.

Examples

`READ:BLUETOOTH:SUMMARY:OUTPUT:POWER:AVERAGE:STATUS?` might return PASS, FAIL, or NA indicating whether the average output power is within the specified limit.

READ:BLUEtooth:SUMMary:OUTPut:POWEr:COUNt? (Query Only)

Returns the number of packets analyzed out of 10 packets for the calculation of the output power from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:OUTPut:POWEr:COUNt?`

Arguments

None

Returns

`<NR1>` the number of packets analyzed.

Examples

`READ:BLUETOOTH:SUMMARY:OUTPUT:POWER:COUNT?` might return 6 indicating that six packets were analyzed.

READ:BLUEtooth:SUMMary:OUTPut:POWEr:EDR:COUNt? (Query Only)

Returns the number of packets analyzed out of 10 packets for calculation of the output power for EDR from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:OUTPut:POWEr:EDR:COUNt?`

Arguments

None

Returns

`<NR1>` the number of packets analyzed.

Examples

`READ:BLUETOOTH:SUMMARY:OUTPUT:POWER:EDR:COUNT?` might return `6` indicating that six packets were analyzed.

READ:BLUEtooth:SUMMary:OUTPut:POWEr:EDR:DPSK? (Query Only)

Returns the power in the DPSK portion of the EDR burst from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:OUTPut:POWEr:EDR:DPSK?`

Arguments

None

Returns

`<NRf>` the power in the DPSK portion of the EDR burst.

Examples

`READ:BLUETOOTH:SUMMARY:OUTPUT:POWER:EDR:DPSK?` might return `-29.2`, indicating that the average output power is -29.2 dBm.

READ:BLUEtooth:SUMMary:OUTPut:POWEr:EDR:GPSK? (Query Only)

Returns the power in the GFSK portion of the EDR burst from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:OUTPut:POWEr:EDR:GPSK?`

Arguments

None

Returns

`<NRf>` the power in the GFSK portion of the EDR burst.

Examples

`READ:BLUETOOTH:SUMMARY:OUTPUT:POWER:EDR:GPSK?` might return `-29.1`, indicating that the average output power is -29.1 dBm.

READ:BLUEtooth:SUMMary:OUTPut:POWEr:PEAK? (Query Only)

Returns the peak output power from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:OUTPut:POWEr:PEAK?`

Arguments

None

Returns

`<NRf>` the peak output power value in dBm.

Examples

`READ:BLUETOOTH:SUMMARY:OUTPUT:POWER:PEAK?` might return `-29.3`, indicating that the peak output power is -29.3 dBm.

READ:BLUEtooth:SUMMary:OUTPut:POWEr:PEAK:STATus? (Query Only)

Returns the Pass/Fail results status of the peak output power from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:OUTPut:POWEr:PEAK:STATus?`

Arguments

None

Returns

`<string>` the Pass/Fail result status.

Examples

`READ:BLUETOOTH:SUMMARY:OUTPUT:POWER:PEAK:STATUS?` might return `PASS`, `FAIL`, or `NA`, indicating whether the peak output power is within the specified limit.

READ:BLUEtooth:SUMMary:PACKet:BDR:CRC? (Query Only)

Returns the Basic Rate value of the CRC field from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:PACKet:BDR:CRC?`

Arguments

None

Returns

`<string>` the value of the CRC field.

Examples

`FETCh:BLUEtooth:SUMMary:PACKet:BDR:CRC?` might return `"0x9DB0"` indicating the value of the CRC field in hexadecimal.

READ:BLUEtooth:SUMMary:PACKet:BDR:HEADer:ARQN? (Query Only)

Returns the value of ARQN field from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:PACKet:BDR:HEADer:ARQN?`

Arguments

None

Returns

`<NR2>` the value of the ARQN field.

Examples

`FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:ARQN?` might return `0.000000`, indicating the value of the flow field.

READ:BLUEtooth:SUMMary:PACKet:BDR:HEADer:FLOW? (Query Only)

Returns the value of Flow field from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:PACKet:BDR:HEADer:FLOW?`

Arguments

None

Returns

`<NR2>` the flow value.

Examples

`FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:FLOW?` might return `1.000000`, indicating the value of the flow field.

READ:BLUEtooth:SUMMary:PACKet:BDR:HEADer:HEC? (Query Only)

Returns the value of HEC field from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:PACKet:BDR:HEADer:HEC?`

Arguments

None

Returns

`<string>` the value of the HEC field.

Examples

`FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:HEC?` might return `"00110010"` indicating the value of the HEC field in binary.

READ:BLUEtooth:SUMMary:PACKet:BDR:HEADer:LT:ADDRess? (Query Only)

Returns the LT address from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:PACKet:BDR:HEADer:LT:ADDRess?`

Arguments

None

Returns

`<string>` the LT address.

Examples

`FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:LT:ADDR?` might return `"101"` indicating the LT address in binary.

READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:SN (Query Only)

Returns the SN value (1 bit) under decoded packet information when the packet type is “OTHERS” for Low Energy in Bluetooth Summary display.

Conditions

Measurement view: Bluetooth Summary display

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:SN?`

Arguments

None

Returns

`<Boolean>` 0, 1

Examples

`READ:BLUETOOTH:SUMMARY:PACKET:BLE:PDU:HEADER:SN` might return 1, indicating the value of SN is "1".

READ:BLUEtooth:SUMMary:PACKet:BDR:HEADer:SEQN? (Query Only)

Returns the value of SEQN field from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:PACKet:BDR:HEADer:SEQN?`

Arguments

None

Returns

`<NR2>` the value of the SEQN field.

Examples

`FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:SEQN?` might return 1.000000 indicating the value of the flow field.

READ:BLUEtooth:SUMMary:PACKet:BDR:HEADer:TYPE? (Query Only)

Returns the value of Type field from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:PACKet:BDR:HEADer:TYPE?`

Arguments

None

Returns

`<string>` the value of the type field.

Examples

`FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:TYPE?` might return `"00001111"` indicating the type field in binary.

READ:BLUEtooth:SUMMary:PACKet:BDR:PAYLoad:LENgth? (Query Only)

Returns the Basic Rate payload length from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:PACKet:BDR:PAYLoad:LENgth?`

Arguments

None

Returns

`<string>` the value of the payload length.

Examples

`FETCh:BLUEtooth:SUMMary:PACKet:BDR:PAYLoad:LENgth?` might return `"0101010011"` indicating the payload length value in binary.

READ:BLUEtooth:SUMMary:PACKet:BDR:PREAmble? (Query Only)

Returns the preamble of the Basic Rate packet from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:PACKet:BDR:PREAmble?`

Arguments

None

Returns

`<string>` the preamble of the packet.

Examples

`FETCh:BLUEtooth:SUMMary:PACKet:BDR:PREAmble?` might return `"0101"` indicating the preamble is 0101 in binary.

READ:BLUEtooth:SUMMary:PACKet:BDR:SYNC:WORD? (Query Only)

Returns the sync word from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:PACKet:BDR:SYNC:WORD?`

Arguments

None

Returns

`<string>` the sync word.

Examples

`FETCh:BLUEtooth:SUMMary:PACKet:BDR:SYNC:WORD?` might return `"Ox4F36F2CEE85390CB"` indicating the sync word in hexadecimal format.

READ:BLUEtooth:SUMMary:PACKet:BDR:TYPE? (Query Only)

Returns the packet type of the Basic Rate signal from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

```
READ:BLUEtooth:SUMMary:PACKet:BDR:TYPE?
```

Arguments

None

Returns

<string> the packet type.

Examples

FETCh:BLUEtooth:SUMMary:PACKet:BDR:TYPE? might return "DH5" indicating that the DH5 packet is being analyzed.

READ:BLUEtooth:SUMMary:PACKet:BLE:ACCEss:ADDRess? (Query Only)

Returns the 32-bit Access Address from the Bluetooth Summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

```
READ:BLUEtooth:SUMMary:PACKet:BLE:ACCEss:ADDRess?
```

Arguments

None

Returns

<string> the 32-bit Access Address.

Examples

READ:BLUETOOTH:SUMMARY:PACKET:BLE:ACCESS:ADDRESS? might return "0x71764129" indicating the Access Address in hexadecimal.

READ:BLUEtooth:SUMMary:PACKet:BLE:CRC? (Query Only)

Returns the value of the CRC field from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:PACKet:BLE:CRC?`

Arguments

None

Returns

`<string>` the value of the CRC field.

Examples

`READ:BLUETOOTH:SUMMARY:PACKET:BLE:CRC?` might return `"0x9DB0"` indicating value CRC field in hexadecimal.

READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:CHANnel:SELect (Query Only)

Returns the channel selected under decoded packet information section in Bluetooth Summary display.

Conditions

Measurement view: Bluetooth Summary display

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:CHANnel:SELect?`

Arguments

None

Returns

`<String>` 0 or 1 or -1 or `"-"`

0 and 1 represents valid

`-1` represents invalid

`"-"` represents NIL

Examples

`READ:BLUETOOTH:SUMMARY:PACKET:BLE:PDU:HEADER:CHANNEL:SELECT` might return 1, indicating the selected channel is valid.

READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:LENGth? (Query Only)

Returns the value of the PDU length field from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:LENGth?`

Arguments

None

Returns

`<string>` the value of the length field.

Examples

`READ:BLUETOOTH:SUMMARY:PACKET:BLE:PDU:HEADER:LENGTH?` might return `"100101"` indicating value length field in binary.

READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:LLID (Query Only)

Returns the LLID value (2 bits) under decoded packet information when the packet type is “OTHERS” for Low Energy in Bluetooth Summary display.

Conditions

Measurement view: Bluetooth Summary display

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:LLID?`

Arguments

None

Returns

`<String>` Either 00,01,10,11

Examples

`READ:BLUETOOTH:SUMMARY:PACKET:BLE:PDU:HEADER:LLID` might return 01, indicating the value of LLID is "01".

READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:MD (Query Only)

Returns the MD value (1 bit) under decoded packet information when the packet type is “OTHERS” for Low Energy in Bluetooth Summary display.

Conditions

Measurement view: Bluetooth Summary display

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:MD?`

Arguments

None

Returns

`<Boolean>` 0, 1

Examples

`READ:BLUETOOTH:SUMMARY:PACKET:BLE:PDU:HEADER:MD` might return 1, indicating the value of MD is "1".

READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:NESN (Query Only)

Returns the NESN value (1 bit) under decoded packet information when the packet type is “OTHERS” for Low Energy in Bluetooth Summary display.

Conditions

Measurement view: Bluetooth Summary display

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:NESN?`

Arguments

None

Returns

`<Boolean>` 0, 1

Examples

`READ:BLUETOOTH:SUMMARY:PACKET:BLE:PDU:HEADER:NESN` might return 1, indicating the value of NESN is "1".

READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:RX:ADDRess? (Query Only)

Returns the PDU Rx address from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:RX:ADDRess?`

Arguments

None

Returns

`<string>` the Rx address.

Examples

`READ:BLUETOOTH:SUMMARY:PACKET:BLE:PDU:HEADER:RX:ADDRESS?` might return 0.00000000 indicating Rx Address.

READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:TX:ADDRess? (Query Only)

Returns the PDU Tx address from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:TX:ADDRess?`

Arguments

None

Returns

`<string>` the Tx address.

Examples

`READ:BLUETOOTH:SUMMARY:PACKET:BLE:PDU:HEADER:TX:ADDRESS?` might return 0.00000000 indicating Tx Address.

READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:TYPE? (Query Only)

Returns the PDU packet type from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:TYPE?`

Arguments

None

Returns

`<string>` the PDU type.

Examples

`READ:BLUETOOTH:SUMMARY:PACKET:BLE:PDU:HEADER:TYPE?` might return `"0010"` indicating the PDU type in binary.

READ:BLUEtooth:SUMMary:PACKet:BLE:PREAmble? (Query Only)

Returns the preamble of the BLE packet from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:PACKet:BLE:PREAmble?`

Arguments

None

Returns

`<string>` the preamble of the packet.

Examples

`READ:BLUETOOTH:SUMMARY:PACKET:BLE:PREAMBLE?` might return `"01010101"` indicating the preamble is 01010101 in binary.

READ:BLUEtooth:SUMMary:PACKet:BLE:TYPE? (Query Only)

Returns the packet type from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Read commands

Syntax

`READ:BLUEtooth:SUMMary:PACKet:BLE:TYPE?`

Arguments

None

Returns

<string> the BLE packet type.

Examples

READ:BLUETOOTH:SUMMARY:PACKET:BLE:TYPE? might return "BLE_TEST" indicating the BLE test signal is being analyzed.

READ:CCDF? (Query Only)

Returns the CCDF measurement results.

Conditions

Measurement views: CCDF

Group

Read commands

Syntax

READ:CCDF?

Arguments

None

Returns

<avg_ampl>,<avg_ccdf>,<crest_factor>,<ampl_10>,<ampl_1>,
<ampl_p1>,<ampl_p01>,<ampl_p001>,<ampl_p0001>

Where

<avg_ampl> is the average amplitude in dBm.

The unit can be changed by the [SENSe]:POWer:UNITs command.

<avg_ccdf> is the average CCDF in percent.

<crest_factor> is the crest factor in dB.

<ampl_10> is the amplitude at CCDF of 10% in dB.

<ampl_1> is the amplitude at CCDF of 1% in dB.

<ampl_p1> is the amplitude at CCDF of 0.1% in dB.

<ampl_p01> is the amplitude at CCDF of 0.01% in dB.

<ampl_p001> is the amplitude at CCDF of 0.001% in dB.

<ampl_p0001> is the amplitude at CCDF of 0.0001% in dB.

Examples

READ:CCDF? might return -33.35,35.8,9.75,3.88,7.07,8.50,9.25,9.72,9.74, indicating

(average amplitude) = -33.35 dBm,

(average CCDF) = 35.8%,

(crest factor) = 9.75 dB,

(amplitude at CCDF of 10%) = 3.88 dB,

(amplitude at CCDF of 1%) = 7.07 dB,

(amplitude at CCDF of 0.1%) = 8.50 dB,

(amplitude at CCDF of 0.01%) = 9.25 dB,

(amplitude at CCDF of 0.001%) = 9.72 dB, and

(amplitude at CCDF of 0.0001%) = 9.74 dB.

READ:CCDF:{FIRSt|SECond|THIRd}:X? (Query Only)

Returns the horizontal values of the specified trace in the CCDF measurement. The mnemonics FIRSt, SECond, and THIRd represent Trace 1, Trace 2, and Gaussian reference curve, respectively.

NOTE: This query is invalid when `[SENSe]:CCDF:TIME:TYPE` is set to CONTinuous or TOTal.

Conditions

Measurement views: CCDF

Group

Read commands

Syntax

`READ:CCDF:{FIRSt|SECond|THIRd}:X?`

Related Commands

READ:CCDF:{FIRSt|SECond|THIRd}[:Y]? (on page 1624)

Arguments

None

Returns

`#<num_digit><num_byte><x(1)><x(2)>...<x(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<x(n)>` is the horizontal value (dB) of the CCDF graph at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`READ:CCDF:FIRST:X` might return #41024xxxx... (1024-byte data) for the horizontal values of Trace 1.

READ:CCDF:{FIRSt|SECond|THIRd}:XY? (Query Only)

Returns the horizontal and vertical value pairs of the specified trace in the CCDF measurement.

The mnemonics FIRSt, SECond, and THIRd represent Trace 1, Trace 2, and Gaussian reference curve, respectively.

NOTE: This query is invalid when `[SENSe]:CCDF:TIME:TYPE` is set to CONTinuous or TOTal.

Conditions

Measurement views: CCDF

Group

Read commands

Syntax

```
READ:CCDF:{FIRSt|SECond|THIRd}:XY?
```

Arguments

None

Returns

#<num_digit><num_byte><x(1)><y(1)><x(2)><y(2)>...<x(n)><y(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<x(n)><y(n)> is the horizontal value (dB) and vertical value (%) pair at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

READ:CCDF:FIRST:XY? might return #41024xxxx... (1024-byte data) for the horizontal and vertical value pairs of Trace 1.

READ:CCDF:{FIRSt|SECond|THIRd}[:Y]? (Query Only)

Returns the vertical values of the specified trace in the CCDF measurement.

The mnemonics FIRSt, SECond, and THIRd represent Trace 1, Trace 2, and Gaussian reference curve, respectively.

NOTE: This query is invalid when [SENSe]:CCDF:TIME:TYPE is set to CONTinuous or TOTal.

Conditions

Measurement views: CCDF

Group

Read commands

Syntax

```
READ:CCDF:{FIRSt|SECond|THIRd}[:Y]?
```

Arguments

None

Returns

#<num_digit><num_byte><y(1)><y(2)>...<y(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<y(n)> is the vertical value (%) of the CCDF graph at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

READ:CCDF:FIRST:Y might return #41024xxxx... (1024-byte data) for the vertical values of Trace 1.

READ:CONSte[:MEASview<y>]:FERRor? (Query only)

Returns the frequency error in Hz. The frequency error is the difference between the measured carrier frequency of the signal and the user selected center frequency of the analyzer.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Constellation

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Constellation).

Group

Read commands

Syntax

READ:CONSte[:MEASview<y>]:FERRor?

Related Commands

READ:EVM[:MEASview<y>]:FERRor? (Query only) (on page 1645)

Arguments

None

Returns

<freq_error>::<NRf> which is the frequency error in Hz.

Examples

READ:CONSte:MEASview2:FERRor? might return -10.7E+3, which is a frequency error of -10.7 kHz for measview 2 of constellation.

READ:CONSte[:MEASview<y>]:RESults? (Query only)

Returns the constellation measurement results of EVM RMS, peak, and location displayed on the bottom of the screen.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Constellation

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Constellation).

Group

Read commands

Syntax

`READ:CONSte[:MEASview<y>]:RESults?`

Arguments

None

Returns

For modulation types 2|4|8|16FSK or C4FM:

`FSK_deviation_Avg_Leftmost`, `FSK_deviation_Avg_Rightmost`

Where

`FSK_deviation_Avg_Leftmost` is the average FSK deviation of the left-most

symbol in Hz.

`FSK_deviation_Avg_Rightmost` is the average FSK deviation of the

right-most symbol in Hz.

For all other valid modulation types:

`<EVM_RMS>,<EVM_peak>,<location>`

Where

`<EVM_RMS>::<NRf>` is the RMS EVM in percent (%).

`<EVM_peak>::<NRf>` is the peak EVM in percent (%).

`<location>::<NRf>` is the peak location in symbol number.

The time unit can be changed by the `[SENSe]:DDEMod:TIMe:UNITs` command.

Examples

`READ:CONSTE:MEASview2:RESULTS?` might return 2.841,3.227,68.000, indicating that the RMS EVM is 2.841% and the peak EVM is 3.227% at symbol #68 for measview 2 of constellation.

READ:CONSte[:MEASview<y>]:TRACe? (Query only)

Returns the constellation trace data.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Constellation

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Constellation).

Group

Read commands

Syntax

`READ:CONSte[:MEASview<y>]:TRACe?`

Arguments

None

Returns

`#<num_digit><num_byte><I(1)><Q(1)><I(2)><Q(2)>...<I(n)><Q(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<I(n)>` and `<Q(n)>` are the normalized I- and Q-coordinate values at the n^{th} data point. 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`READ:CONSTE:MEASview2:TRACE?` might return #43848xxxx... (3848-byte data) for the constellation trace data for measview 2 of constellation.

READ:DDEMod[:MEASview<y>]:STABle? (Query only)

Returns the symbol table data.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Symbol table

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Symbol table).

Group

Read commands

Syntax

`READ:DDEMod[:MEASview<y>]:STABle?`

Arguments

None

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the symbol table data for the point n, \4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`READ:DDEMod:MEASview1:STABle?` might return #3512xxxx... (512-byte data) for the symbol Table measurement view1.

READ:DIQVtime[:MEASview<y>]:FERRor? (Query only)

Returns the frequency error in the Demod I&Q versus Time measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Demod I&Q versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod I&Q versus Time).

Group

Read commands

Syntax

`READ:DIQVtime:[MEASview<y>]:FERRor?`

Arguments

None

Returns

`<freq_error>::<NRf>` is the frequency error in Hz.

Examples

`READ:DIQVtime:MEASview1:FERROR?` might return -10.7E+3, indicating the frequency error is -10.7 kHz measurement view 1.

READ:DIQVtime[:MEASview<y>]:I? (Query only)

Returns the I versus Time trace data.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Demod I&Q versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod I&Q versus Time).

Group

Read commands

Syntax

`READ:DIQVtime[:MEASview<y>]:I?`

Arguments

None

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the I level in volts at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

READ:DIQVtime:MEASview1:I? might return #3160xxxx... (160-byte data) for the I versus Time trace measurement view 1.

READ:DIQVtime[:MEASview<y>]:Q? (Query only)

Returns the Q versus Time trace data.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Demod I&Q versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod I&Q versus Time).

Group

Read commands

Syntax

READ:DIQVtime[:MEASview<y>]:Q?

Arguments

None

Returns

#<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the Q level in volts at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

READ:DIQVtime:MEASview1:Q? might return #3160xxxx... (160-byte data) for the Q versus Time trace measurement view 1.

READ:DPX:DDENsity? (Query Only)

Returns the calculated density for the selected region in the DPXogram measurement.

Conditions

Measurement views: DPX, Spectrum

Group

Read commands

Syntax
`READ:DPX:DDENsity?`

Arguments
None

Returns
`<NR1>` value.

Examples
`READ:DPX:DDENSITY?` might return a DPX signal density value of 24.4802413393.

READ:DPX:DGRam:LINE:COUNt? (Query Only)
Returns the number of lines in the DPXogram measurement.

Conditions
Measurement views: DPX, Spectrum

Group
Read commands

Syntax
`READ:DPX:DGRam:LINE:COUNt?`

Arguments
None

Returns
`<NR1>` number of lines in the measurement.

Examples
`READ:DPX:DGRAM:LINE:COUNT?` might return `40` indicating that there were 40 lines in the measurement.

READ:DPX:DGRam:TIME[:SCALe]:OFFSet? (Query Only)
Returns the time scale offset in the DPXogram measurement.

Conditions
Measurement views: DPX, Spectrum

Group
Read commands

Syntax
`READ:DPX:DGRam:TIME[:SCALe]:OFFSet?`

Arguments

None

Returns

<string> The time offset in the DPXogram measurement in seconds.

Examples

READ:DPX:DGRAM:TIME[:SCALE]:OFFSET? might return "0.000 s" indicating that the time scale offset was 0.000 s.

READ:DPX:RESults:TRACe<x>? (Query Only)

Acquires a waveform and then returns waveform data of the specified trace <x> in the DPX spectrum measurement, where x is 1 to 6. The traces 1 to 4 are in the standard form. Trace 5 is the bitmap trace and its data is returned in a binary block. Trace 6 is the DPXogram trace on DPX spectrum plots. Trace 7 is the Ogram line in the DPXogram display.

Conditions

Measurement views: DPX, Spectrum

Group

Read commands

Syntax

READ:DPX:RESults:TRACe<x>?

Arguments

<NR1>

Returns

For traces 1 to 4 or 6:

#<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the amplitude of the trace sample at the nth data point.

<data(n) is in the 4-byte little endian floating-point format specified in IEEE 488.2.

The format is a binary block of 32 bit floating point values. Each value ranges from 0 to 1.0, where 1.0 represents 100%.

Examples

READ:DPX:RESULTS:TRACE1? might return #42004xxxx... (2004-byte of data) for the waveform data of trace one (1).

READ:DPX:TRACe:AVERage? (Query Only)

Acquires a waveform and then returns waveform data of the average trace in the DPX, Spectrum measurement.

Conditions

Measurement views: DPX, Spectrum

Group

Read commands

Syntax

`READ:DPX:TRACe:AVERage?`

Arguments

None

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the waveform data of the average trace for the point n in dBm, 4-byte little endian floating-point format specified in IEEE 488.2.

The unit can be changed by the `[SENSe]:POWer:UNITs` command.

Examples

`READ:DPX:TRACE:AVERAGE?` might return #42004xxxx... (2004-byte data) for the waveform data of the average trace.

READ:DPX:TRACe:BITMap? (Query Only)

Acquires a waveform and then returns trace waveform data of the bitmap trace in the DPX, Spectrum measurement.

Conditions

Measurement views: DPX, Spectrum

Group

Read commands

Syntax

`READ:DPX:TRACe:BITMap?`

Arguments

None

Returns

The format is a binary block of 32 bit floating point values. Each value ranges from 0 to 1.0, where 1.0 represents 100%.

Examples

`READ:DPX:TRACE:BITMAP?` might return `#42004xxxx...` (2004-byte data) for the waveform data of the bitmap trace.

READ:DPX:TRACe:MATH? (Query Only)

Returns waveform data of the math trace in the DPX, Spectrum measurement.

Conditions

Measurement views: DPX, Spectrum

Group

Read commands

Syntax

```
READ:DPX:TRACe:MATH?
```

Arguments

None

Returns

<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the waveform data of the math trace for the point n in dBm, 4-byte little endian floating-point format specified in IEEE 488.2.

The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples

READ:DPX:TRACE:MATH? might return #42004xxxx... (2004-byte data) for the waveform data of the math trace.

READ:DPX:TRACe:MAXimum? (Query Only)

Returns waveform data of the maximum trace in the DPX, Spectrum measurement.

Conditions

Measurement views: DPX, Spectrum

Group

Read commands

Syntax

```
READ:DPX:TRACe:MAXimum?
```

Arguments

None

Returns

<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the waveform data of the maximum trace for the point n in dBm, 4-byte little endian floating-point format specified in IEEE 488.2.

The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples

READ:DPX:TRACE:MAXIMUM? might return #42004xxxx... (2004-byte data) for the waveform data of the maximum trace.

READ:DPX:TRACe:MINimum? (Query Only)

Returns waveform data of the minimum trace in the DPX, Spectrum measurement.

Conditions

Measurement views: DPX, Spectrum

Group

Read commands

Syntax

READ:DPX:TRACe:MINimum?

Arguments

None

Returns

#<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the waveform data of the minimum trace for the point n in dBm. 4-byte little endian floating-point format specified in IEEE 488.2. The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples

READ:DPX:TRACE:MINIMUM? might return #42004xxxx... (2004-byte data) for the waveform data of the minimum trace.

READ:EDIagram[:MEASview<y>]:FDEViation? (Query only)

Returns the frequency deviation versus Time trace data with the X values.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Group

Read commands

Syntax

READ:EDIagram[:MEASview<y>]:FDEViation?

Returns

#<num_digit><num_byte><Y(1)><X(1)><Y(2)><X(2)>...<Y(n)><X(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<Y(n)> and <X(n)> is frequency deviation in Hz and time (symbols) pair at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

READ:EDIagram:MEASview2:FDEVation? might return #3160xxxx... (160-byte data) for the frequency deviation versus Time trace for measview2 of the Eye diagram.

READ:EDIagram[:MEASview<y>]:FERRor? (Query only)

Returns the frequency error in the eye diagram measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Eye diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Eye diagram).

Group

Read commands

Syntax

READ:EDIagram[:MEASview<y>]:FERRor?

Arguments

None

Returns

<freq_error>::<NRf> is the frequency error in Hz.

Examples

READ:EDIagram:MEASview2:FERRor? might return -10.7E+3, indicating the frequency error is -10.7 kHz for measview2 of the Eye diagram.

READ:EDIagram[:MEASview<y>]:I? (Query only)

Returns the I versus Time trace data.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Eye diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Eye diagram).

Group

Read commands

Syntax

READ:EDIagram[:MEASview<y>]:I?

Arguments

None

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<Y(n)><X(n)>` is the I level (normalized) and time (symbols) pair at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`READ:EDIagram:MEASview2:I?` might return #3160xxxx... (160-byte data) for the I versus Time trace for measview2 of the Eye diagram.

READ:EDIagram[:MEASview<y>]:Q? (Query only)

Returns the Q versus Time trace data.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Eye diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Eye diagram).

Group

Read commands

Syntax

`READ:EDIagram[:MEASview<y>]:Q?`

Arguments

None

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<Y(n)><X(n)>` is the Q level (normalized) and time (symbols) pair at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`READ:EDIagram:MEASview2:Q?` might return #3160xxxx... (160-byte data) for the Q versus Time trace for measview2 of the Eye diagram.

READ:EMC:COUNt? (Query Only)

Returns the total numbers of spots detected in EMC View.

Conditions

Measurement view: EMC - EMI Display

Group

Read commands

Syntax

`READ:EMC:COUNt?`

Arguments

None

Returns

`<NR1>`

Examples

`READ:EMC:COUNT?` might return `12`, indicating that 12 spots are detected.

READ:EMC:EMISsion:RESults:STATus? (Query Only)

Returns the status of the emission results in EMC view.

Conditions

Measurement view: EMC - EMI Display

Group

Read commands

Syntax

`READ:EMC:EMISsion:RESults:STATus?`

Arguments

None

Returns

`<String>` either PASS or FAIL

Examples

`READ:EMC:EMISSION:RESULTS:STATUS?` might return `FAIL`, indicating that the status of emission results is failed.

READ:EMC:SPOT<x>:AMBient:AMPLitude? (Query Only)

Returns the amplitude of the selected spot in ambient trace of EMC View.

<x> = the spot number

Conditions

Measurement view: EMC - EMI Display

Group

Read commands

Syntax

`READ:EMC:SPOT<x>:AMBIent:AMPLitude?`

Arguments

None

Returns

`<NRf>`

Examples

`READ:EMC:SPOT1:AMBIent:AMPLitude?` might return `52.612`, indicating that amplitude of spot 1 in ambient trace is 52.61 dBuV.

READ:EMC:SPOT<x>:AMBIent:DELTa? (Query Only)

Returns the Delta value of the selected spot in ambient trace of EMC View.

<x> = the spot number

Conditions

Measurement view: EMC - EMI Display

Group

Read commands

Syntax

`READ:EMC:SPOT<x>:AMBIent:DELTa?`

Arguments

None

Returns

`<NRf>`

Examples

`READ:EMC:SPOT<x>:AMBIent:DELTa?` might return `3.3614`, indicating that delta value of spot 2 in the ambient trace is 3.36 dBuV.

READ:EMC:SPOT<x>:AMBIent:PASS? (Query Only)

Returns the status of the selected spot in ambient trace in EMC view. The default status is THRE.

<x> = the spot number

Conditions

Measurement view: EMC - EMI Display

Group

Read commands

Syntax

`READ:EMC:SPOT<x>:AMBIent:PASS?`

Arguments

None

Returns

`<String>` Either OLIM, OMAR or THRE

Where

`OLIM` = Over Limit

`OMAR` = Over Margin

`THRE` = Threshold

Examples

`READ:EMC:SPOT3:AMBIent:PASS?` might return `THRE`, indicating that the spot 3 of ambient trace is failing by crossing Threshold.

READ:EMC:SPOT<x>:FREQuency? (Query Only)

Returns the frequency of the specified spot in EMC View.

<x> = <NR1> Indicates the spot number

Conditions

Measurement view: EMC - EMI Display

Group

Read commands

Syntax

`READ:EMC:SPOT<x>:FREQuency?`

Arguments

None

Returns

`<NRf>`

Examples

`READ:EMC:SPOT2:FREQuency?` might return `366.12500000E+3`, indicating the frequency of spot 2 is 366.125 kHz.

READ:EMC:SPOT<x>:PASS? (Query Only)

Returns the overall status of the specified spot in EMC view. The default status is THRE.

<x> = <NR1> Indicates the spot number

Conditions

Measurement view: EMC - EMI Display

Group

Read commands

Syntax

`READ:EMC:SPOT<x>:PASS?`

Arguments

None

Returns

`<String>` Either OLIM, OMAR or THRE

Where

`OLIM` = Over Limit

`OMAR` = Over Margin

`THRE` = Threshold

Examples

`READ:EMC:SPOT1:PASS?` might return `OLIM`, indicating that spot 1 is failing by crossing the Over Limit.

READ:EMC:SPOT<x>:RANGe? (Query Only)

Returns the range of the selected spot in EMC View.

<x> = <NR1> Indicates the spot number

Conditions

Measurement view: EMC - EMI Display

Group

Read commands

Syntax

`READ:EMC:SPOT<x>:RANGe?`

Arguments

None

Returns

`<String>` from A to T

Examples

`READ:EMC:SPOT1:RANGe?` might return `"B"`, indicating that the range of spot 1 is "B".

READ:EMC:SPOT<x>:RECAll:AMPLitude? (Query Only)

Returns the amplitude of the specified spot in the recalled trace of EMC View.

<x> = Indicates the spot number

Conditions

Measurement view: EMC - EMI Display

Group

Read commands

Syntax

`READ:EMC:SPOT<x>:RECAll:AMPLitude?`

Arguments

None

Returns

`<NRf>`

Examples

`READ:EMC:SPOT1:RECAll:AMPLitude?` might return `23.598`, indicating that amplitude of spot 1 in the recalled trace is 23.59 dBuV.

READ:EMC:SPOT<x>:RECAll:DELTa? (Query Only)

Returns the Delta value of the selected spot in recalled trace of EMC View.

<x> = the spot number

Conditions

Measurement view: EMC - EMI Display

Group

Read commands

Syntax

`READ:EMC:SPOT<x>:RECAll:DELTa?`

Arguments

None

Returns

`<NRf>`

Examples

`READ:EMC:SPOT2:RECAll:DELTa?` might return `5.5878`, indicating that delta value of spot 2 in the recalled trace is 5.58 dBuV.

READ:EMC:SPOT<x>:RECAll:PASS? (Query Only)

Returns the status of the specified spot in recalled trace in EMC view. The default status is THRE.

<x> = the spot number

Conditions

Measurement view: EMC - EMI Display

Group

Read commands

Syntax

`READ:EMC:SPOT<x>:RECAll:PASS?`

Arguments

None

Returns

`<String>` Either OLIM, OMAR or THRE

Where

`OLIM` = Over Limit

`OMAR` = Over Margin

`THRE` = Threshold

Examples

`READ:EMC:SPOT<x>:RECAll:PASS?` might return `OMAR`, indicating that the spot 1 of recalled trace is failing by crossing Margin.

READ:EMC:SPOT<x>:SCAN<y>:AMPLitude? (Query Only)

Returns the amplitude of the specified spot with respect to specified scan detector in the results table.

<x> = the spot number

<y> = the scan detector number (either 1 or 2)

Conditions

Measurement view: EMC - EMI Display

Group

Read commands

Syntax

`READ:EMC:SPOT<x>:SCAN<y>:AMPLitude?`

Arguments

None

Returns

`<NRf>`

Examples

`READ:EMC:SPOT1:SCAN1:AMPLitude?` might return `20.6209`, indicating that amplitude of spot 1 for scan detector 1 is 20.62 dBuV.

READ:EMC:SPOT<x>:SCAN<y>:DELTa? (Query Only)

Returns the Delta value of the specified spot with respect to the specified scan detector in the results table.

<x> = the spot number

<y> = the scan detector number (either 1 or 2)

Conditions

Measurement view: EMC - EMI Display

Group

Read commands

Syntax

`READ:EMC:SPOT<x>:SCAN<y>:DELTa?`

Arguments

None

Returns

`<NRf>`

Examples

`READ:EMC:SPOT2:SCAN1:DELTa` might return `3.6209`, indicating that delta value of spot 2 for scan detector 1 is 3.62 dBuV.

READ:EMC:SPOT<x>:SCAN<y>:PASS? (Query Only)

Returns the status of the specified spot based on the specified scan detector in EMC view. The default status is THRE.

<x> = the spot number

<y> = the scan detector number (either 1 or 2)

Conditions

Measurement view: EMC - EMI Display

Group

Read commands

Syntax

`READ:EMC:SPOT<x>:SCAN<y>:PASS?`

Arguments

None

Returns

`<String>` Either OLIM, OMAR or THRE

Where

`OLIM` = Over Limit

`OMAR` = Over Margin

THRE = Threshold

Examples

READ:EMC:SPOT2:SCAN1:PASS? might return THRE, indicating that spot 2 of scan detector 1 is failing by crossing the threshold.

READ:EMC:SPOT<x>:SPOT<y>:AMPLitude? (Query Only)

Returns the amplitude of the specified spot with respect to specified spot detector in the results table.

<x> = the spot number

<y> = the spot detector number (either 1, 2, or 3)

Conditions

Measurement view: EMC - EMI Display

Group

Read commands

Syntax

READ:EMC:SPOT<x>:SPOT<y>:AMPLitude?

Arguments

None

Returns

<NRf>

Examples

READ:EMC:SPOT1:SPOT1:AMPLitude? might return 18.365, indicating that the amplitude of spot 1 for spot detector 1 is 18.36 dBuV.

READ:EMC:SPOT<x>:SPOT<y>:DELTa? (Query Only)

Returns the Delta value of the specified spot with respect to specified spot detector in the results table.

<x> = the spot number

<y> = the spot detector number (either 1, 2, or 3)

Conditions

Measurement view: EMC - EMI Display

Group

Read commands

Syntax

READ:EMC:SPOT<x>:SPOT<y>:DELTa?

Arguments

None

Returns

<NRf>

Examples

READ:EMC:SPOT2:SPOT1:DELTa? might return 2.2578, indicating that delta value of spot 2 for spot detection 1 is 2.25 dBuV.

READ:EMC:SPOT<x>:SPOT<y>:PASS? (Query Only)

Returns the status of the specified spot based on the specified spot detector in EMC view. The default status is THRE.

<x> = the spot number

<y> = the spot detector number (either 1, 2, or 3)

Conditions

Measurement view: EMC - EMI Display

Group

Read commands

Syntax

READ:EMC:SPOT<x>:SPOT<y>:PASS?

Arguments

None

Returns

<String> Either OLIM, OMAR or THRE

Where

OLIM = Over Limit

OMAR = Over Margin

THRE = Threshold

Examples

READ:EMC:SPOT1:SPOT2:PASS? might return OMAR, indicating that the spot 1 of spot detector 2 is failing by crossing the Margin.

READ:EVM[:MEASview<y>]:FERRor? (Query only)

Returns the frequency error in the EVM versus Time measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: EVM versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod EVM versus Time).

Group

Read commands

Syntax

`READ:EVM:[MEASVIEW<y>]:FERRor?`

Arguments

None

Returns

`<freq_error>::<NRf>` is the frequency error in Hz.

Examples

`READ:EVM:MEASVIEW2:FERROR?` might return `-10.7E+3`, indicating the frequency error is -10.7 kHz for measview2 of EVM.

READ:EVM[:MEASview<y>]:PEAK? (Query only)

Returns the peak value in the EVM versus Time measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: EVM versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod EVM versus Time).

Group

Read commands

Syntax

`READ:EVM[:MEASview<y>]:PEAK?`

Related Commands

READ:EVM[:MEASview<y>]:PINDex? (on page 1646)

Arguments

None

Returns

`<peak>::<NRf>` is the peak EVM value in percent (%).

Examples

`READ:EVM:MEASview2:PEAK?` might return 1.32, indicating the peak EVM value is 1.32% for measview2 of EVM.

READ:EVM[:MEASview<y>]:PINDex? (Query only)

Returns the time at the EVM peak.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: EVM versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod EVM versus Time).

Group

Read commands

Syntax

`READ:EVM[:MEASview<y>]:PINDex?`

Related Commands

READ:EVM[:MEASview<y>]:PEAK? (on page 1646)

Arguments

None

Returns

`<peak_time>::<NRf>` is the time at the EVM peak in symbol number. The unit can be changed by the `[SENSe]:DDEMod:TIMe:UNITs` command.

Examples

`READ:EVM:MEASview2:PINDEX?` might return 68.000, indicating that the EVM peak is at symbol #68 for measview2 of EVM.

READ:EVM[:MEASview<y>]:RMS? (Query only)

Returns the RMS (Root-Mean-Square) value in the EVM versus Time measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: EVM versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod EVM versus Time).

Group

Read commands

Syntax

`READ:EVM[:MEASview<y>]:RMS?`

Arguments

None

Returns

`<rms>::<NRf>` is the RMS EVM value in percent (%).

Examples

`READ:EVM:MEASview2:RMS?` might return `0.582`, indicating the RMS EVM value is 0.582% for measview2 of EVM.

READ:EVM[:MEASview<y>]:TRACe? (Query only)

Returns the EVM versus Time trace data.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: EVM versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod EVM versus Time).

Group

Read commands

Syntax

`READ:EVM[:MEASview<y>]:TRACe?`

Arguments

None

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the EVM versus Time trace data for the point n in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`READ:EVM:MEASVIEW2:TRACE?` might return #42036xxxx... (2036-byte data) for the EVM versus Time measview2.

READ:FDVTime[:MEASview<y>]:FERRor? (Query only)

Returns the frequency error in the Frequency deviation versus Time measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Frequency deviation versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Frequency deviation versus Time).

Group

Read commands

Syntax

`READ:FDVTime:MEASview<y>:FERRor?`

Arguments

None

Returns

<freq_error>::<NRf> is the frequency error in Hz.

Examples

READ:FDVTIME:MEASview2:FERROR? might return –10.7E+3, indicating the frequency error is -10.7 kHz.

READ:FDVTime[:MEASview<y>]:TRACe? (Query only)

Returns the Frequency deviation versus Time trace data.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Frequency deviation versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Frequency deviation versus Time).

Group

Read commands

Syntax

READ:FDVTime:MEASview<y>:TRACe?

Arguments

None

Returns

#<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the Frequency deviation versus Time trace data for the point n in Hz, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

READ:FDVTime:MEASview2:TRACE? might return #3160xxxx... (160-byte data) for the Frequency deviation versus Time measview 2.

READ:{FM|PM}:FERRor? (Query only)

Returns the frequency error in the Frequency modulation and Phase modulation measurements.

Conditions

Measurement views: Frequency and Phase modulation

Group

Read commands

Syntax

`READ:{FM|PM}:FERRor?`

Arguments

None

Returns

`<freq_error>::<NRf>` is the frequency error in Hz.

Examples

`READ:FM:FERRor?` might return `-10.7E+3`, indicating the frequency error is -10.7 kHz.

READ:FM:PHALf? (Query Only)

Returns the half peak-peak frequency deviation (Pk-Pk/2) in the FM measurement.

Conditions

Measurement views: Frequency deviation versus Time

Group

Read commands

Syntax

`READ:FM:PHALf?`

Arguments

None

Returns

`<Pk-Pk/2>::<NRf>` is the half peak-peak frequency deviation in Hz.

Examples

`READ:FM:PHALF?` might return `628.9E+3`, indicating the half peak-peak frequency deviation is 628.9 kHz.

READ:FM:PNEGative? (Query Only)

Returns the negative peak frequency deviation (−Pk) in the FM measurement.

Conditions

Measurement views: FM

Group

Read commands

Syntax

`READ:FM:PNEGative?`

Arguments

None

Returns

<-Pk>::<NRf> is the negative peak frequency deviation in Hz.

Examples

READ:FM:PNEGATIVE? might return -495.6E+3, indicating the negative peak frequency deviation is -495.6 kHz.

READ:FM:PPOSitive? (Query Only)

Returns the positive peak frequency deviation (+Pk) in the FM measurement.

Conditions

Measurement views: FM

Group

Read commands

Syntax

READ:FM:PPOSitive?

Arguments

None

Returns

<+Pk>::<NRf> is the positive peak frequency deviation in Hz.

Examples

READ:FM:PPOSITIVE? might return 763.2E+3, indicating the positive peak frequency deviation is 763.2 kHz.

READ:FM:PTPeak? (Query Only)

Returns the peak-peak frequency deviation (Pk-Pk) in the FM measurement.

Conditions

Measurement views: FM

Group

Read commands

Syntax

READ:FM:PTPeak?

Arguments

None

Returns

<Pk-Pk>::<NRf> is the peak-peak frequency deviation in Hz.

Examples

`READ:FM:PTPEAK?` might return `1.258E+6`, indicating the peak-peak frequency deviation is 1.258 MHz.

READ:FM:RESult? (Query Only)

Returns the FM measurement results.

Conditions

Measurement views: FM

Group

Read commands

Syntax

`READ:FM:RESult?`

Arguments

None

Returns

`<+Pk>,<-Pk>,<RMS>,<Pk-Pk>,<Pk-Pk/2>`

Where

`<+Pk>::<NRf>` is the positive peak frequency deviation in Hz.

`<-Pk>::<NRf>` is the negative peak frequency deviation in Hz.

`<RMS>::<NRf>` is the RMS frequency deviation in Hz.

`<Pk-Pk>::<NRf>` is the peak-peak frequency deviation in Hz.

`<Pk-Pk/2>::<NRf>` is the half peak-peak frequency deviation in Hz.

Examples

`READ:FM:RESULT?` might return `763.2E+3,-494.6E+3,271.2E+3,1.258E+6,628.9E+3`.

READ:FM:RMS? (Query Only)

Returns the RMS frequency deviation in the FM measurement.

Conditions

Measurement views: FM

Group

Read commands

Syntax

`READ:FM:RMS?`

Arguments

None

Returns

<RMS>::<NRf> is the RMS frequency deviation in Hz.

Examples

READ:FM:RMS? might return 271.2E+3, indicating the RMS frequency deviation is 271.2 kHz.

READ:{FSETtling|PSETtling}:FTTime? (Query Only)

Returns the settling time from the trigger position in seconds. The settling time is measured from the trigger point (see :FETCh:FSETtling:TRIGger:TIME).

Conditions

Measurement views: Frequency and Phase Settling

Group

Read commands

Syntax

READ:{FSETtling|PSETtling}:FTTime?

Arguments

None

Returns

<value>::<NRf> is the settling time in seconds.

Examples

READ:FSETTLING:FTTIME? might return 44.8300E-6, indicating the settling time is 44.83 µs.

READ:{FSETtling|PSETtling}:MASK[:PASS]? (Query Only)

Returns whether the input signal passes the mask test.

Conditions

Measurement views: Frequency and Phase Settling

Group

Read commands

Syntax

READ:{FSETtling|PSETtling}:MASK[:PASS]?

Arguments

None

Returns

0 is returned if the signal fails the mask test.

1 is returned if the signal passes the mask test.

Examples

`READ:FSETTLING:MASK:PASS?` might return `1`, indicating the signal passed (did not exceed the mask limits).

READ:{FSETtling|PSETtling}:SETTled:FREQuency? (Query Only)

Returns the frequency at which the signal is considered settled.

Conditions

Measurement views: Frequency and Phase Settling

Group

Read commands

Syntax

`READ:{FSETtling|PSETtling}:SETTled:FREQuency?`

Arguments

None

Returns

`<value>::<NRf>` is the frequency at which the signal is settled.

Examples

`READ:FSETTLING:SETTLED:FREQUENCY?` might return `2.44838E+9`, indicating the input signal frequency at the point where the signal is considered settled is 2.44838 GHz.

READ:{FSETtling|PSETtling}:SETTled[:PASS]? (Query Only)

Returns whether the input signal is settled with the tolerance range.

Conditions

Measurement views: Frequency and Phase Settling

Group

Read commands

Syntax

`READ:{FSETtling|PSETtling}:SETTled[:PASS]?`

Arguments

None

Returns

0 is returned if the signal is not settled within the tolerance range.

1 is returned if the signal is settled within the tolerance range.

Examples

`READ:FSETTLING:SETTLED:PASS?` might return `1`, indicating the signal is settled within the tolerance range.

READ:{FSETtling|PSETtling}:SETTled:TIME? (Query Only)

Returns the settled time in seconds. The settled time is measured from the measurement start point.

Conditions

Measurement views: Frequency and Phase Settling

Group

Read commands

Syntax

`READ:{FSETtling|PSETtling}:SETTled:TIME?`

Arguments

None

Returns

`<value>::<NRf>` is the settled time in seconds.

Examples

`READ:FSETTLING:SETTLED:TIME?` might return `299.830000E-6`, indicating the settled time is 299.83 µs.

READ:{FSETtling|PSETtling}:SLMSd[:PASS]? (Query Only)

Returns whether the input signal is settled within the specified tolerance and the signal is settled longer than the Minimum Settled Duration `([SENSe]:{FSETtling|PSETtling}:SDURation:MINimum)`.

Conditions

Measurement views: Frequency and Phase Settling

Group

Read commands

Syntax

`READ:{FSETtling|PSETtling}:SLMSd[:PASS]?`

Arguments

None

Returns

0 is returned if the signal is not settled within the tolerance range and minimum settled duration..

1 is returned if the signal is settled within the tolerance range and minimum settled duration.

Examples

`READ:FSETTLING:SLMSD:PASS?` might return `1`, indicating the signal is settled within the tolerance range and the minimum settled duration.

READ:{FSETtling|PSETtling}:STARt:TIME? (Query Only)

Returns the start time for the measurement in seconds. The start time is measured from the start of the analysis period.

Conditions

Measurement views: Frequency and Phase Settling

Group

Read commands

Syntax

`READ:{FSETtling|PSETtling}:STARt:TIME?`

Arguments

None

Returns

`<value>::<NRf>` is the time in seconds when the measurement started.

Examples

`READ:FSETTLING:START:TIME?` might return `251.4300E-6`, indicating the measurement started 251.43 µs after the beginning of the analysis period.

READ:{FSETtling|PSETtling}:TIME? (Query Only)

Returns the settling time in seconds. The settling time is measured from the start time (see :FETCh:FSETtling:STARt:TIME (on page 1164)).

Conditions

Measurement views: Frequency and Phase Settling

Group

Read commands

Syntax

`READ:{FSETtling|PSETtling}:TIME?`

Arguments

None

Returns

`<value>::<NRf>` is the settling time in seconds.

Examples

`READ:FSETTLING:TIME?` might return `48.4000E-6`, indicating the settling time is 48.4 µs.

READ:{FSETtling|PSETtling}:TRACe<x>:X? (Query Only)

Returns the Frequency or Phase values of the specified trace.

The parameter `<x>` = 1 and 2, representing Trace 1 and Trace 2, respectively.

Conditions

Measurement views: Frequency and Phase Settling

Group

Read commands

Syntax

`READ:{FSETtling|PSETtling}:TRACe<x>:X?`

Arguments

None

Returns

`#<num_digit><num_byte><y(1)><y(2)>...<y(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<y(n)>` is the frequency (Hz) or phase (degrees) at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`READ:FSETTLING:TRACE1:X?` might return #574232xxxx... (74232-byte data) for the frequency values of Trace 1.

READ:{FSETtling|PSETtling}:TRACe<x>:XY? (Query Only)

Returns the time and frequency or phase value pairs of the specified trace.

The parameter `<x>` = 1 and 2, representing Trace 1 and Trace 2, respectively.

Conditions

Measurement views: Frequency and Phase Settling

Group

Read commands

Syntax

`READ:{FSETtling|PSETtling}:TRACe<x>:XY?`

Arguments

None

Returns

`#<num_digit><num_byte><x(1)><y(1)><x(2)><y(2)>...<x(n)><y(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<x(n)><y(n)>` is the time and frequency or phase value pair at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`READ:FSETTLING:TRACE2:XY?` might return #574232xxxx... (74232-byte data) for the time and frequency pairs of the Trace 2.

READ:{FSETtling|PSETtling}:TRACe<x>[:Y]? (Query Only)

Returns the frequency or phase values of the specified trace.

The parameter `<x>` = 1 and 2, representing Trace 1 and Trace 2, respectively.

Conditions

Measurement views: Frequency and Phase Settling

Group

Read commands

Syntax

```
READ:{FSETtling|PSETtling}:TRACe<x>[:Y]?
```

Arguments

None

Returns

```
#<num_digit><num_byte><y(1)><y(2)>...<y(n)>
```

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<y(n)>` is the frequency or phase value at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`READ:FSETTLING:TRACE1:Y?` might return #575148xxxx... (75148-byte data) for the frequency values of Trace 1.

READ:{FSETtling|PSETtling}:TRIGger:TIME? (Query Only)

Returns the time when the trigger occurred in seconds.

Conditions

Measurement views: Frequency and Phase Settling

Group

Read commands

Syntax

```
READ:{FSETtling|PSETtling}:TRIGger:TIME?
```

Arguments

None

Returns

`<value>::<NRf>` is the time in seconds after the measurement start point when the trigger occurred.

Examples

`READ:FSETTLING:TRIGGER:TIME?` might return `2.255E-6`, indicating the time when the trigger occurred 255 µs after the start of the measurement.

READ:{FSETtling|PSETtling}:VALue? (Query Only)

Returns the settled value in Hz for Frequency Settling and in degrees for Phase Settling.

Conditions

Measurement views: Frequency and Phase Settling

Group

Read commands

Syntax

`READ:{FSETtling|PSETtling}:VALue?`

Arguments

None

Returns

`<value>::<NRf>` is the settling value in Hz for Frequency Settling and in degrees for Phase Settling.

Examples

`READ:FSETTLING:VALUE?` might return `2.44838155E+9`, indicating the settled frequency is 2.44838 GHz.

READ:FSETtling:ERRor? (Query Only)

Returns the settled error in Hz in Frequency Settling. Only for Frequency Settling Time measurement. In Frequency Settling:

When Target Reference is set to Auto, Settled Error = 0.

Conditions

Measurement views: Frequency Settling

Group

Read commands

Syntax

`READ:FSETtling:ERRor?`

Arguments

None

Returns

`<value>::<NRf>` returns the settled error in Hz.

When Target Reference is set to Auto, Settled Error = 0.

When Target Reference is set to Meas Freq:

Settled Error = Settled Frequency – (Measurement Frequency + Offset)

Examples

`READ:FSETTLING:ERRor?` might return 0, indicating the Target Reference is set to Auto.

READ:FVTime[:MEASview<y>]? (Query Only)

Returns the Frequency versus Time trace data.

Conditions

Measurement views: Frequency versus Time

Group

Read commands

Syntax

`READ:FVTime[:MEASview<y>]?`

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the frequency drift data for the point n in Hz, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`READ:FVTIME:MEASview1?` might return `#3156xxxx...` (156-byte data) for the Frequency versus Time trace.

READ:FVTime[:MEASview<y>]:MAXimum? (Query Only)

Returns the maximum value in the Frequency versus Time measurement.

Conditions

Measurement views: Frequency versus Time

Group

Read commands

Syntax

`READ:FVTime[:MEASview<y>]:MAXimum?`

Related Commands

READ:FVTime[:MEASview<y>]:MAXLocation? (on page 1661)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

<max>::<NRf> is the maximum frequency drift in Hz.

Examples

READ:FVTIME:MEASview1:MAXIMUM? might return 2.625E+6, indicating the maximum frequency drift is 2.625 MHz.

READ:FVTime[:MEASview<y>]:MAXLocation? (Query Only)

Returns the time at which the frequency drift is maximum.

Conditions

Measurement views: Frequency versus Time

Group

Read commands

Syntax

READ:FVTime[:MEASview<y>]:MAXLocation?

Related Commands

READ:FVTime[:MEASview<y>]:MAXimum? (on page 1660)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

<max_time>::<NRf> is the time in seconds at which the frequency drift is maximum.

Examples

READ:FVTIME:MEASview1:MAXLOCATION? might return 25.03E-9, indicating the frequency drift is maximum at 25.03 ns.

READ:FVTime[:MEASview<y>]:MINimum? (Query Only)

Returns the minimum value in the Frequency versus Time measurement.

Conditions

Measurement views: Frequency versus Time

Group

Read commands

Syntax

READ:FVTime[:MEASview<y>]:MINimum?

Related Commands

READ:FVTime[:MEASview<y>]:MINLocation? (on page 1662)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

<min>::<NRf> is the minimum frequency drift in Hz.

Examples

READ:FVTIME:MEASview1:MINIMUM? might return -6.618E+6, indicating the minimum frequency drift is -6.618 MHz.

READ:FVTime[:MEASview<y>]:MINLocation? (Query Only)

Returns the time at which the frequency drift is minimum.

Conditions

Measurement views: Frequency versus Time

Group

Read commands

Syntax

READ:FVTime[:MEASview<y>]:MINLocation?

Related Commands

READ:FVTime[:MEASview<y>]:MINimum? (on page 1661)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

<min_time>::<NRf> is the time in seconds at which the frequency drift is minimum.

Examples

READ:FVTIME:MEASview1:MINLOCATION? might return 450.7E-9, indicating the frequency drift is minimum at 450.7 ns.

READ:FVTime[:MEASview<y>]:OGRAm? (Query only)

Returns the ogram trace data. The line is selected using the TRACe:FVTime:MEASview<y>:OGRAm:SELect:LINE command.

Conditions

Applicable measurement view must be active.

Group

Read commands

Syntax

`READ:FVTime[:MEASview<y>]:OGRAm?`

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional. By default y is 1.

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in <num_byte>.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the frequency at the n^{th} data point in Hz, represented as a 4-byte little-endian IEEE floating-point value.

The unit can be changed by the `[SENSe]:POWer:UNITs` command.

Examples

`READ:FVTime:MEASview1:OGRAm?` might return #43204xxxx... (3204-byte data) for the ogram trace.

READ:FVTime[:MEASview<y>]:RESult? (Query Only)

Returns the Frequency versus Time measurement results.

Conditions

Measurement views: Frequency versus Time

Group

Read commands

Syntax

`READ:FVTime[:MEASview<y>]:RESult?`

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

<max>,<max_time>,<min>,<min_time>

Where

`<max>::<NRf>` is the maximum frequency drift in Hz.

`<max_time>::<NRf>` is the time in seconds at which the frequency drift is maximum.

`<min>::<NRf>` is the minimum frequency drift in Hz.

`<min_time>::<NRf>` is the time in seconds at which the frequency drift is minimum.

Examples

`READ:FVTIME:MEASview1:RESULT?` might return `2.625E+6,25.03E-9,-6.618E+6,450.7E-9`, indicating the maximum frequency drift is 2.625 MHz at 25.03 ns and the minimum frequency drift is -6.618 MHz at 450.7 ns.

READ:FVTime[:MEASview<y>]:TRACe:BITMap? (Query Only)

Acquires a waveform and then returns trace waveform data of the bitmap trace in the Frequency versus Time measurement.

Conditions

Measurement views: Frequency versus Time

Group

Read commands

Syntax

`READ:FVTime[:MEASview<y>]:TRACe:BITMap?`

Arguments

None

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Returns

The format is a binary block of 32 bit floating point values. Each value ranges from 0 to 1.0, where 1.0 represents 100%.

Examples

`READ:FVTIME:TRACE:BITMAP?` might return #42004xxxx... (2004-byte data) for the waveform data of the bitmap trace.

READ:IQVTime[:MEASview<y>]:I? (Query Only)

Returns the I versus Time trace data.

Conditions

Measurement views: RF I & Q versus Time

Group

Read commands

Syntax

`READ:IQVTime[:MEASview<y>]:I?`

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the I level data for the point n in volts, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

READ:IQVTIME:MEASview1:I? might return #3160xxxx... (160-byte data) for the I versus Time trace.

READ:IQVTime[:MEASview<y>]:MAXimum? (Query Only)

Returns the maximum value in the RF I & Q versus Time measurement.

Conditions

Measurement views: RF I & Q versus Time

Group

Read commands

Syntax

READ:IQVTime[:MEASview<y>]:MAXimum?

Related Commands

READ:IQVTime[:MEASview<y>]:MAXLocation? (on page 1665)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

<max>::<NRf> is the maximum I or Q level in volts.

Examples

READ:IQVTIME:MEASview1:MAXIMUM? might return 1.214, indicating the maximum I or Q level is 1.214 V.

READ:IQVTime[:MEASview<y>]:MAXLocation? (Query Only)

Returns the time at which the I or Q level is maximum.

Conditions

Measurement views: RF I & Q versus Time

Group

Read commands

Syntax

READ:IQVTime[:MEASview<y>]:MAXLocation?

Related Commands

READ:IQVTime[:MEASview<y>]:MAXimum? (on page 1665)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

<max_time>::<NRf> is the time in seconds at which the I or Q level is maximum.

Examples

READ:IQVTIME:MEASview1:MAXLOCATION? might return 175.3E-9, indicating the I or Q level is maximum at 175.3 ns.

READ:IQVTime[:MEASview<y>]:MINimum? (Query Only)

Returns the minimum value in the RF I & Q versus Time measurement.

Conditions

Measurement views: RF I & Q versus Time

Group

Read commands

Syntax

```
READ:IQVTime[:MEASview<y>]:MINimum?
```

Related Commands

READ:IQVTime[:MEASview<y>]:MINLocation? (on page 1666)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

<min>::<NRf> is the minimum I or Q level in volts.

Examples

READ:IQVTIME:MEASview1:MINIMUM? might return -370.5E-3, indicating the minimum I or Q level is -370.5 mV.

READ:IQVTime[:MEASview<y>]:MINLocation? (Query Only)

Returns the time at which the I or Q level is minimum.

Conditions

Measurement views: RF I & Q versus Time

Group

Read commands

Syntax

```
READ:IQVTime[:MEASview<y>]:MINLocation?
```

Related Commands

READ:IQVTime[:MEASview<y>]:MINimum? (on page 1666)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

<min_time>::<NRf> is the time in seconds at which the I or Q level is minimum.

Examples

READ:IQVTIME:MEASview1:MINLOCATION? might return 450.7E-9, indicating the I or Q level is minimum at 450.7 ns.

READ:IQVTime[:MEASview<y>]:Q? (Query Only)

Returns the Q versus Time trace data.

Conditions

Measurement views: IQ versus Time

Group

Read commands

Syntax

READ:IQVTime[:MEASview<y>]:Q?

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

#<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the Q level data for the point n in volts, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

READ:IQVTIME:MEASview1:Q? might return #3160xxxx... (160-byte data) for the Q versus Time trace.

READ:IQVTime[:MEASview<y>]:RESult? (Query Only)

Returns the RF I & Q versus Time measurement results.

Conditions

Measurement views: RF I & Q versus Time

Group

Read commands

Syntax

`READ:IQVTime[:MEASview<y>]:RESult?`

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

`<max>,<max_time>,<min>,<min_time>`

Where

`<max>::<NRf>` is the maximum I or Q level in volts.

`<max_time>::<NRf>` is the time in seconds at which the I or Q level is

maximum.

`<min>::<NRf>` is the minimum I or Q level in volts.

`<min_time>::<NRf>` is the time in seconds at which the I or Q level is minimum.

Examples

`READ:IQVTIME:MEASview1:RESULT?` might return `1.214,175.3E-9,-370.5E-3,450.7E-9`, indicating the maximum I or Q level is 1.214 V at 175.3 ns and the minimum I or Q level is -370.5 mV at 450.7 ns.

READ:LTE:ACLR:MHITs:COUNt? (Query Only)

Returns the number of rows in the results table of the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

Group

Read commands

Syntax

`READ:LTE:ACLR:MHITs:COUNt?`

Returns

`<NR1>`

Examples

`READ:LTE:ACLR:MHITS:COUNT?` might return `7`, indicating there are seven rows in the results table.

READ:LTE:ACLR:MHITs<x>:BANDwidth? (Query Only)

Returns the bandwidth value of the channel (row) of the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

The parameter <x> is the row number in the LTE ACLR display results table.

Group

Read commands

Syntax

`READ:LTE:ACLR:MHITs<x>:BANDwidth?`

Returns

`<NR3>`

Examples

`READ:LTE:ACLR:MHITS1:BANDWIDTH?` might return `4.5150000000E+6`, which indicates a bandwidth for channel 1 (row 1) of 4.515 MHz.

READ:LTE:ACLR:MHITs<x>:CHANnel:NAME? (Query Only)

Returns the channel name of the specified index in the results table of the LTE ACLR display. The minimum index will be 1 and the maximum index will be the count of the channels in the results table.

Conditions

Measurement view: LTE ACLR

The parameter <x> is a positive integer and represents the row number in the LTE ACLR display results table.

Group

Read commands

Syntax

`READ:LTE:ACLR:MHITs<x>:CHANnel:NAME?`

Returns

`<string>`

Examples

`READ:LTE:ACLR:MHITS2:CHANNEL:NAME?` might return `"B"`, which is the name of channel 2 (row 2).

READ:LTE:ACLR:MHITs<x>:FREQuency:OFFSet? (Query Only)

Returns the frequency offset of the channel of the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

The parameter <x> is a positive integer and represents the row number in the LTE ACLR display results table.

Group

Read commands

Syntax

`READ:LTE:ACLR:MHITs<x>:FREQuency:OFFSet?`

Returns

`<NR3>`

Examples

`READ:LTE:ACLR:MHITS1:FREQUENCY:OFFSET?` might return `5.0000000000E+6`, which indicates the frequency offset of channel 1 (row 1) is 5 MHz.

READ:LTE:ACLR:MHITs<x>:INTeg:LOWEr:ABSolute? (Query Only)

Returns the lower absolute value of the channel of the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

The parameter <x> is the row number in the LTE ACLR display results table.

Group

Read commands

Syntax

`READ:LTE:ACLR:MHITs<x>:INTeg:LOWEr:ABSolute?`

Returns

`<NRf>`

Examples

`READ:LTE:ACLR:MHITs1:INTeg:LOWEr:ABSolute?` might return `-53.3921980303` which indicates lower absolute value of channel 1 (row 1) is -53.3921980303 dBm.

NOTE: The actual units depends on the choice made from the units tab of the Analysis control panel.

READ:LTE:ACLR:MHITs<x>:INTeg:LOWEr:RELative? (Query Only)

Returns the lower relative value of the channel of the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

The parameter <x> is the row number in the LTE ACLR display results table.

Group

Read commands

Syntax

`READ:LTE:ACLR:MHITs<x>:INTeg:LOWEr:RELative?`

Returns

`<NRf>`

Examples

`READ:LTE:ACLR:MHITS1:INTEG:LOWER:RELATIVE?` might return `-53.3921980303`, which indicates lower relative value of channel 1 (row 1) is -53.3921980303 dB.

READ:LTE:ACLR:MHITs<x>:INTeg:UPPEr:ABSolute? (Query Only)

Returns the upper absolute value of the channel of the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

The parameter <x> is a positive integer and represents the row number in the LTE ACLR display results table.

Group

Read commands

Syntax

`READ:LTE:ACLR:MHITs<x>:INTeg:UPPEr:ABSolute?`

Returns

`<NRf>`

Examples

`READ:LTE:ACLR:MHITS1:INTEG:UPPER:ABSOLUTE?` might return `-53.3921980303`, which indicates upper absolute value of channel 1 (row 1) is -53.3921980303 dBm.

NOTE: The actual units depends on the choice made from the units tab of the Analysis control panel.

READ:LTE:ACLR:REFerence:POWer? (Query Only)

Returns the reference power level in the LTE ACLR measurement.

Conditions

Measurement view: LTE ACLR

Group

Read commands

Syntax

`READ:LTE:ACLR:REFerence:POWer?`

Returns

`<NRf>`

Examples

`READ:LTE:ACLR:REFERENCE:POWER?` might return `-16.92`, indicating that the reference power level is –16.92 dBm.

READ:LTE:ACLR:RESUlts:STATus? (Query Only)

Returns the pass or fail status of the LTE ACLR display measurement.

Conditions

Measurement view: LTE ACLR

Group

Read commands

Syntax

`READ:LTE:ACLR:RESUlts:STATus?`

Returns

0 means measurement status is FAIL.

1 means measurement status is PASS.

Examples

`READ:LTE:ACLR:RESULTS:STATUS?` might return `1`, indicating that the LTE ACLR measurement status is PASS.

READ:LTE:ACLR:SPECtrum:X? (Query Only)

Returns the frequency of the spectrum trace in the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

Group

Read commands

Syntax

`READ:LTE:ACLR:SPECtrum:X?`

Returns

`#<num_digit><num_byte><x(1)><x(2)>...<x(n)>`

Where,

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<x(n)>` is the frequency (Hz) at the nth point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`READ:LTE:ACLR:SPECTRUM:X?` might return `#516020xxxx...` (16020-byte data) for the frequencies of the spectrum trace.

READ:LTE:ACLR:SPECtrum:XY? (Query Only)

Returns the frequency and amplitude pairs of the spectrum trace in the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

Group

Read commands

Syntax

`READ:LTE:ACLR:SPECtrum:XY?`

Returns

#<num_digit><num_byte><x(1)><y(1)><x(2)><y(2)>...<x(n)><y(n)>

Where,

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<x(n)><y(n)> is the frequency (Hz) and amplitude (dBm) pair at the nth point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

READ:LTE:ACLR:SPECTRUM:XY? might return #516020xxxx... (16020-byte data) for the frequency and amplitude pairs of the spectrum trace.

READ:LTE:ACLR:SPECtrum:Y? (Query Only)

Returns the amplitude of the spectrum trace in the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

Group

Read commands

Syntax

READ:LTE:ACLR:SPECtrum:Y?

Returns

#<num_digit><num_byte><y(1)><y(2)>...<y(n)>

Where,

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<y(n)> is the frequency (Hz) at the nth point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

READ:LTE:ACLR:SPECTRUM:Y? might return #516020xxxx... (16020-byte data) for the amplitude of the spectrum trace.

READ:LTE:CHSPectrum:OBW? (Query Only)

Returns the Occupied Bandwidth (Hz) of the LTE Channel Spectrum measurement view.

Conditions

Measurement view: LTE Channel Spectrum

Group

Read commands

Syntax

READ:LTE:CHSPectrum:OBW?

Returns

<NR3>

Examples

READ:LTE:CHSPECTRUM:OBW? might return 1.0851851250E+6, indicating the occupied Bandwidth is 1.09 MHz.

READ:LTE:CHSPectrum:POWer:CHANnel? (Query Only)

Returns the Channel Power value in LTE Channel Spectrum display. The result will be in dBm (by default) or in the units chosen from the Units tab of the Analysis Control panel.

Conditions

Measurement view: LTE Channel Spectrum

Group

Read commands

Syntax

READ:LTE:CHSPectrum:POWer:CHANnel?

Returns

<NRf>

Examples

READ:LTE:CHSPECTRUM:POWER:CHANNEL? might return -14.9248560147, indicating that the Channel Power is -14.92 dBm.

READ:LTE:CHSPectrum:SPECtrum? (Query Only)

Returns spectrum trace data of the LTE Channel Spectrum measurement.

Conditions

Measurement view: LTE Channel Spectrum

Group

Read commands

Syntax

READ:LTE:CHSPectrum:SPECtrum?

Returns

#<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the amplitude in dBm at the nth data point, 4-byte little endian floating point format specified in IEEE 488.2.

Examples

`READ:LTE:CHSPECTRUM:SPECTRUM?` might return `#43204 xxxx...` (3204-byte data) for the spectrum trace data of the LTE Channel Spectrum measurement.

READ:LTE:CONSte:CELL:ID? (Query Only)

Queries the Cell ID value LTE Constellation display.

Conditions

Measurement view: LTE Constellation

Group

Read commands

Syntax

`READ:LTE:CONSte:CELL:ID?`

Returns

`<NR1>`

Examples

`READ:LTE:CONSTE:CELL:ID?` might return `112`, indicating CELLID measurement number is 112.

READ:LTE:CONSte:FREQuency:ERRor? (Query Only)

Queries Frequency Error value in Hz for the LTE Constellation display.

Conditions

Measurement view: LTE Constellation

Group

Read commands

Syntax

`READ:LTE:CONSte:FREQuency:ERRor?`

Returns

`<NRf>`

Examples

`READ:LTE:CONSTE:FREQUENCY:ERROR?` might return `-71.9780578613`, indicating frequency error value is -71.9780578613 Hz.

READ:LTE:CONSte:GROUp:ID? (Query Only)

Queries the Group ID value for the LTE Constellation display.

Conditions

Measurement view: LTE Constellation

Group

Read commands

Syntax
`READ:LTE:CONSte:GROUp:ID?`

Returns
`<NR1>`

Examples
`READ:LTE:CONSTE:GROUP:ID?` might return `160`, indicating Group ID measurement number is 160.

READ:LTE:CONSte:SECTor:ID? (Query Only)
Queries the Sector ID value for the LTE Constellation display.

Conditions
Measurement view: LTE Constellation

Group
Read commands

Syntax
`READ:LTE:CONSte:SECTor:ID?`

Returns
`<NRf>`

Examples
`READ:LTE:CONSTE:SECTOR:ID?` might return `2`, indicating the Sector ID value is 2.

READ:LTE:PVTime:OFFSlot:POWer? (Query Only)
Returns the TOff power measurement value in dBm/MHz for the LTE Power vs Time display.

Conditions
Measurement view: LTE Power vs Time

Group
Read commands

Syntax
`READ:LTE:PVTime:OFFSlot:POWer?`

Returns
`<NRf>`

Examples
`READ:LTE:PVTIME:OFFSLOT:POWER?` might return `-76.11514587403`, indicating the offslot power is -76.11514 dBm/MHz.

READ:LTE:PVTime:RESUlts:STATus

Returns the pass or fail status for the LTE Power vs Time measurement.

Conditions

Measurement view: LTE Power vs Time

Group

Read commands

Syntax

`READ:LTE:PVTime:RESUlts:STATus`

Returns

0: means measurement status is FAIL.

1: means measurement status is PASS.

Examples

`READ:LTE:PVTIME:RESULTS:STATUS` might return `1`, indicating the measurement status is PASS.

READ:LTE:PVTime:TRACe:X? (Query Only)

Returns the horizontal values (time in seconds) for the LTE Power vs. Time trace.

Conditions

Measurement view: LTE Power vs Time

Group

Read commands

Syntax

`READ:LTE:PVTime:TRACe:X?`

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where,

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the nth point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`READ:LTE:PVTIME:TRACE:X?` might return `#43204` (3204-byte data), which represents the time in seconds (horizontal values).

READ:LTE:PVTime:TRACe:XY? (Query Only)

Returns the horizontal value (time in seconds) and vertical value (power) for the LTE Power vs. Time trace.

Conditions

Measurement view: LTE Power vs Time

Group

Read commands

Syntax

`READ:LTE:PVTime:TRACe:XY?`

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where,

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the nth point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`READ:LTE:PVTIME:TRACE:XY?` might return `#43204` (3204-byte data), which represents the horizontal value (time in seconds) and vertical value (power) pair at the nth data point.

READ:LTE:PVTime:TRACe:Y? (Query Only)

Returns the vertical values (power) for the LTE Power vs. Time trace.

Conditions

Measurement view: LTE Power vs Time

Group

Read commands

Syntax

`READ:LTE:PVTime:TRACe:Y?`

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where,

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the nth point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`READ:LTE:PVTIME:TRACE:Y?` might return `#43204` (3204-byte data), which represents the vertical values (power).

READ:MCPower:ADJacent:CHANnels? (Query Only)

Returns the power of adjacent channels in order of increasing frequency.

Conditions

Measurement views: MCPR

Group

Read commands

Syntax

`READ:MCPower:ADJacent:CHANnels?`

Arguments

None

Returns

`<acpr_lower(n)>,...<acpr_lower(2)>,<acpr_lower(1)>,`
`<acpr_upper(1)>,<acpr_upper(2)>,...<acpr_upper(n)>`

Where

`<acpr_lower(n)>` is the ACPR for the lower channel #n in dB.

`<acpr_upper(n)>` is the ACPR for the upper channel #n in dB.

To add a pair of upper and lower adjacent channels, use the `[SENSe]:MCPower:CHANnel:ADJacent:ADD` command.

Examples

`READ:MCPOWER:ADJACENT:CHANNELS?` might return

`-4.420,-4.847,-4.316,-4.225`, indicating

(ACPR for the lower channel 2) = -4.420 dB,

(ACPR for the lower channel 1) = -4.847 dB,

(ACPR for the upper channel 1) = -4.316 dB, and

(ACPR for the upper channel 2) = -4.225 dB.

READ:MCPower:CHANnel:POWer? (Query Only)

Returns the reference power in the MCPR measurement.

Conditions

Measurement views: MCPR

Group

Read commands

Syntax

`READ:MCPower:CHANnel:POWer?`

Arguments

None

Returns

`<ref_power>:<NRf>` is the reference power in dBm. The unit can be changed by the `[SENSe]:POWer:UNITs` command.

To select the power reference, use the `[SENSe]:MCPower:RCHannels` commands.

Examples

`READ:MCPOWER:CHANNEL:POWER?` might return `4.227`, indicating that the reference power is 4.227 dBm.

READ:MCPower:MAIN:CHANnels? (Query Only)

Returns the power of main channels in order of increasing frequency.

Conditions

Measurement views: MCPR

Group

Read commands

Syntax

`READ:MCPower:MAIN:CHANnels?`

Related Commands

[SENSe]:MCPower:CHANnel:MAIN:{BANDwidth|BWIDth} (on page 2183)

[SENSe]:MCPower:CHANnel:MAIN:COUNt (on page 2184)

[SENSe]:MCPower:CHANnel:MAIN:INACtive (on page 2184)

[SENSe]:MCPower:CHANnel:MAIN:SPACing (on page 2185)

Arguments

None

Returns

`<power_main(1)>,<power_main(2)>,...<power_main(n)>`

Where

`<power_main(n)>` is the power of main channel #n in dBm. The unit can be changed by the `[SENSe]:POWer:UNITs` command.

To specify the main channels, use the `[SENSe]:MCPower:CHANnel:MAIN` commands.

Examples

`READ:MCPOWER:MAIN:CHANNELS?` might return

`-2.420,-2.847,-2.316,-2.225`, indicating

(power of the main channel 1) = -2.420 dBm,

(power of the main channel 2) = -2.847 dBm,

(power of the main channel 3) = -2.316 dBm, and

(power of the main channel 4) = -2.225 dBm.

READ:MCPower:SPECtrum? (Query Only)

Returns spectrum trace data of the MCPR measurement.

Conditions

Measurement views: MCPR

Group

Read commands

Syntax

`READ:MCPower:SPECtrum?`

Arguments

None

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the spectrum trace data in dBm for the point n, 4-byte little endian floating-point format specified in IEEE 488.2.

The unit can be changed by the `[SENSe]:POWer:UNITs` command.

Examples

`READ:MCPOWER:SPECTRUM?` might return `#43204xxxx...` (3204-byte data) for the spectrum trace data of the MCPR measurement.

READ:MERRor[:MEASview<y>]:FERRor? (Query only)

Returns the frequency error in the Magnitude error versus Time measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Magnitude error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Magnitude error versus Time).

Group

Read commands

Syntax

`READ:MERRor[:MEASview<y>]:FERRor?`

Arguments

None

Returns

`<freq_error>::<NRf>` is the frequency error in Hz.

Examples

`READ:MERRor:MEASview2:FERRor?` might return -10.7E+3, indicating the frequency error is -10.7 kHz.

READ:MERRor[:MEASview<y>]:PEAK? (Query only)

Returns the peak value in the Magnitude error versus Time measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Magnitude error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Magnitude error versus Time).

Group

Read commands

Syntax

`READ:MERRor[:MEASview<y>]:PEAK?`

Related Commands

READ:MERRor[:MEASview<y>]:PINDex? (on page 1682)

Arguments

None

Returns

`<peak>::<NRf>` is the peak magnitude error value in percent (%).

Examples

`READ:MERRor:MEASview2:PEAK?` might return 1.57, indicating the peak magnitude error is 1.57%.

READ:MERRor[:MEASview<y>]:PINDex? (Query only)

Returns the time at the Magnitude error peak.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Magnitude error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Magnitude error versus Time).

Group

Read commands

Syntax

`READ:MERRor[:MEASview<y>]:PINDex?`

Related Commands

READ:MERRor[:MEASview<y>]:PEAK? (on page 1682)

Arguments

None

Returns

<peak_time>::<NRf> is the time at the Magnitude error peak in symbol number. The unit can be changed by the [SENSe]:DDEMod:TIMe:UNITs command.

Examples

READ:MERRor:MEASview2:PINDex? might return 68.000, indicating that the Magnitude error peak is at symbol #68.

READ:MERRor[:MEASview<y>]:RMS? (Query only)

Returns the RMS (Root-Mean-Square) value in the Magnitude error versus Time measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Magnitude error versus Time.

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Magnitude error versus Time).

Group

Read commands

Syntax

READ:MERRor[:MEASview<y>]:RMS?

Arguments

None

Returns

<rms>::<NRf> is the RMS magnitude error value in percent (%).

Examples

READ:MERRor:MEASview2:RMS? might return 0.382, indicating the RMS MERRor value is 0.382%.

READ:MERRor[:MEASview<y>]:TRACe? (Query only)

Returns the Magnitude error versus Time trace data.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Magnitude error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Magnitude error versus Time).

Group

Read commands

Syntax

`READ:MERRor[:MEASview<y>]:TRACe?`

Arguments

None

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the Magnitude error versus Time trace data for the point n in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`READ:MERRor:MEASview2:TRACe?` might return #42036xxxx... (2036-byte data) for the Magnitude error versus Time measview 2.

READ:NOISe:TABLe:VALue? (Query Only)

Returns the noise table data. A total of 7 values per frequency (28 bytes per frequency) is returned. The number of frequencies returned depends on the settings.

Conditions

Measurement view: Noise display

Group

Read commands

Syntax

`READ:NOISe:TABLe:VALue?`

Returns

Returns an array of floats (size of 4 bytes each) defined as:

- First frequency gain, noise figure, noise factor, y factor, noise temperature,cpower hot, power cold
- Second frequency gain, noise figure, noise factor, y factor, noise temperature, power hot, power cold

Examples

`READ:NOISE:TABLE:VALUE?`

READ:OBWidth:BOBW:XDBBandwidth? (Query Only)

Returns the difference between the higher and lower frequency points corresponding to the value where it is X dB less from the value at the center frequency.

Set the measurement direction using the appropriate inward or outward command before issuing this command.

Conditions

Measurement views: Bluetooth 20 dB Bandwith

Group

Read commands

Syntax

```
READ:OBWidth:BOBW:XDBBandwidth?
```

Related Commands

READ:OBWidth:BOBW:XDBMbandwidth:IN:BANDwidth? (on page 1685)

READ:OBWidth:BOBW:XDBMbandwidth:OUT:BANDwidth? (on page 1687)

Arguments

None

Returns

`<value> ::= <NRf>` the X dBm bandwidth in Hz.

Examples

`READ:OBWIDTH:BOBW:XDBBANDWIDTH?` might return `89.12000E+3`, indicating the difference between the higher and lower frequency points is 89.12 kHz.

READ:OBWidth:BOBW:XDBMbandwidth:IN:BANDwidth? (Query Only)

Returns the difference between the higher and lower frequency points corresponding to the value which it is the X dBm value measured inwards (coming from the ends to the center frequency).

Conditions

Measurement views: Bluetooth 20 dB Bandwith

Group

Read commands

Syntax

```
READ:OBWidth:BOBW:XDBMbandwidth:IN:BANDwidth?
```

Related Commands

READ:OBWidth:BOBW:XDBBandwidth? (on page 1684)

READ:OBWidth:BOBW:XDBMbandwidth:IN:RIGHt:FREQuency? (on page 1686)

Arguments

None

Returns

`<value> ::= <NRf>` the input X dBm value in Hz.

Examples

READ:OBWIDTH:BOBW:XDBMBANDWIDTH:IN:BANDWIDTH? might return 1.2518400000E+6, indicating that the bandwidth corresponding to the input XdBm vaue is 1.2518 MHz measured in the inward direction.

READ:OBWidth:BOBW:XDBMbandwidth:IN:LEFT:FREQuency? (Query Only)

Returns the frequency corresponding to the X dBm value to the left of the center frequency measured in the inward direction.

Conditions

Measurement views: Bluetooth 20 dB Bandwith

Group

Read commands

Syntax

READ:OBWidth:BOBW:XDBMbandwidth:IN:LEFT:FREQuency?

Related Commands

READ:OBWidth:BOBW:XDBMbandwidth:IN:RIGHt:FREQuency? (on page 1686)

Arguments

None

Returns

<value> ::= <NRf> the left frequency in Hz.

Examples

READ:OBWIDTH:BOBW:XDBMBANDWIDTH:IN:LEFT:FREQUENCY? might return 2.401E+9, indicating that the frequency corresponding to the X dbm value to the left of the center frequency is 2.401 GHz.

READ:OBWidth:BOBW:XDBMbandwidth:IN:RIGHt:FREQuency? (Query Only)

Returns the frequency corresponding to the X dBm value to the right of the center frequency measured in the inward direction.

Conditions

Measurement views: Bluetooth 20 dB Bandwith

Group

Read commands

Syntax

READ:OBWidth:BOBW:XDBMbandwidth:IN:RIGHt:FREQuency?

Related Commands

READ:OBWidth:BOBW:XDBMbandwidth:IN:LEFT:FREQuency? (on page 1686)

Arguments

None

Returns

`<value> ::= <NRf>` the right frequency in Hz.

Examples

`READ:OBWIDTH:BOBW:XDBMBANDWIDTH:IN:RIGHT:FREQUENCY?` might return `2.445E+9` indicating that the frequency corresponding to the XdBm value to the right of the center the frequency is 2.445 GHz.

READ:OBWidth:BOBW:XDBMbandwidth:OUT:BANDwidth? (Query Only)

Returns the difference between the higher and lower frequency points corresponding to the value which it is the X dBm value measured outwards (from the center frequency to the ends).

Conditions

Measurement views: Bluetooth 20 dB Bandwith

Group

Read commands

Syntax

`READ:OBWidth:BOBW:XDBMbandwidth:OUT:BANDwidth?`

Related Commands

READ:OBWidth:BOBW:XDBBandwidth? (on page 1684)

READ:OBWidth:BOBW:XDBMbandwidth:IN:BANDwidth? (on page 1685)

Arguments

None

Returns

`<value> ::= <NRf>` the input X dBm value in Hz.

Examples

`READ:OBWIDTH:BOBW:XDBMBANDWIDTH:OUT:BANDWIDTH?` might return `1.2518400000E+6` indicating that the bandwidth corresponding to the output XdBm vaue is 1.2518 MHz measured in the outward direction.

READ:OBWidth:BOBW:XDBMbandwidth:OUT:LEFT:FREQuency? (Query Only)

Returns the frequency corresponding to the X dBm value to the left of the center frequency measured in the outward direction.

Conditions

Measurement views: Bluetooth 20 dB Bandwith

Group

Read commands

Syntax

```
READ:OBWidth:BOBW:XDBMbandwidth:OUT:LEFT:FREQuency?
```

Related Commands

READ:OBWidth:BOBW:XDBMbandwidth:OUT:RIGHt:FREQuency? (on page 1688)

Arguments

None

Returns

`<value> ::= <NRf>` the left frequency in Hz.

Examples

`READ:OBWIDTH:BOBW:XDBMBANDWIDTH:OUT:LEFT:FREQUENCY?` might return `2.401E+9` indicating that the frequency corresponding to the X dbm value to the left of the center frequency is 2.401 Hz.

READ:OBWidth:BOBW:XDBMbandwidth:OUT:RIGHt:FREQuency? (Query Only)

Returns the frequency corresponding to the X dBm value to the right of the center frequency measured in the outward direction.

Conditions

Measurement views: Bluetooth 20 dB Bandwith

Group

Read commands

Syntax

```
READ:OBWidth:BOBW:XDBMbandwidth:OUT:RIGHt:FREQuency?
```

Related Commands

READ:OBWidth:BOBW:XDBMbandwidth:OUT:LEFT:FREQuency? (on page 1687)

Arguments

None

Returns

`<value> ::= <NRf>` the Right frequency in Hz.

Examples

`READ:OBWIDTH:BOBW:XDBMBANDWIDTH:OUT:RIGHT:FREQUENCY?` might return `2.445E+9` indicating that the frequency corresponding to the XdBm value to the right of the center the frequency is 2.445 GHz.

READ:OBWidth:FREQuency:ERRor? (Query Only)

Returns the frequency error in the Occupied Bandwidth measurement.

Conditions

Measurement views: Occupied Bandwidth

Group

Read commands

Syntax

`READ:OBWidth:FREQuency:ERRor?`

Arguments

None

Returns

`<freq_error>::<NRf>` is the frequency error in Hz.

Examples

`READ:OBWIDTH:FREQUENCY:ERROR?` might return `-10.7E+3`, indicating the frequency error is -10.7 kHz.

READ:OBWidth:OBWidth:BANDwidth? (Query Only)

Returns the occupied bandwidth in the Occupied Bandwidth measurement.

Conditions

Measurement views: Occupied Bandwidth

Group

Read commands

Syntax

READ:OBWidth:OBWidth:BANDwidth? (on page 1689)

Arguments

None

Returns

`<OBW>::<NRf>` is the occupied bandwidth in Hz.

Examples

`READ:OBWIDTH:OBWIDTH:BANDWIDTH?` might return `4.0E+6`, indicating the occupied bandwidth is 4 MHz.

READ:OBWidth:OBWidth:LEFT:FREQuency? (Query Only)

Returns the left (lower) frequency of the occupied bandwidth.

Conditions

Measurement views: Occupied Bandwidth

Group

Read commands

Syntax

```
READ:OBWidth:OBWidth:LEFT:FREQuency?
```

Related Commands

READ:OBWidth:OBWidth:RIGHt:FREQuency? (on page 1691)

Arguments

None

Returns

<OBW_left_freq>::<NRf> is the left frequency in Hz.

Examples

```
READ:OBWIDTH:OBWIDTH:LEFT:FREQUENCY?
```

READ:OBWidth:OBWidth:LEFT:LEVel? (Query Only)

Returns the level at the left frequency of the occupied bandwidth.

Conditions

Measurement views: Occupied Bandwidth

Group

Read commands

Syntax

READ:OBWidth:OBWidth:LEFT:LEVel?

Related Commands

READ:OBWidth:OBWidth:RIGHt:LEVel? (on page 1691)

Arguments

None

Returns

<OBW_left_level>::<NRf> is the level at the left frequency in dB.

Examples

`READ:OBWIDTH:OBWIDTH:LEFT:LEVEL?` might return `-23.5`, indicating the level at the left frequency is -23.5 dB.

READ:OBWidth:OBWidth:POWer? (Query Only)

Returns the reference power in the Occupied Bandwidth measurement.

Conditions

Measurement views: Occupied Bandwidth

Group

Read commands

Syntax

`READ:OBWidth:OBWidth:POWer?`

Arguments

None

Returns

`<OBW_ref_power>::<NRf>` is the reference power in dBm.

The unit can be changed by the `[SENSe]:POWer:UNITs` command.

Examples

`READ:OBWIDTH:OBWIDTH:POWER?` might return `-10.0`, indicating the reference power is -10 dBm.

READ:OBWidth:OBWidth:RIGHt:FREQuency? (Query Only)

Returns the right (higher) frequency of the occupied bandwidth.

Conditions

Measurement views: Occupied Bandwidth

Group

Read commands

Syntax

`READ:OBWidth:OBWidth:RIGHt:FREQuency?`

Related Commands

READ:OBWidth:OBWidth:LEFT:FREQuency? (on page 1689)

Arguments

None

Returns

`<OBW_right_freq>::<NRf>` is the right frequency in Hz.

Examples

`READ:OBWIDTH:OBWIDTH:RIGHT:FREQUENCY?` might return `1.502E+9`, indicating the right frequency is 1.502 GHz.

READ:OBWidth:OBWidth:RIGHt:LEVel? (Query Only)

Returns the level at the right frequency of the occupied bandwidth.

Conditions

Measurement views: Occupied Bandwidth

Group

Read commands

Syntax

```
READ:OBWidth:OBWidth:RIGHt:LEVel?
```

Related Commands

READ:OBWidth:OBWidth:LEFT:LEVel? (on page 1690)

Arguments

None

Returns

<OBW_right_level>::<NRf> is the level at the right frequency in dB.

Examples

READ:OBWIDTH:OBWIDTH:RIGHT:LEVEL? might return -23.5, indicating the level at the right frequency is -23.5 dB.

READ:OBWidth:SPECtrum? (Query Only)

Returns spectrum trace data of the Occupied Bandwidth measurement.

Conditions

Measurement views: Occupied Bandwidth

Group

Read commands

Syntax

```
READ:OBWidth:SPECtrum?
```

Arguments

None

Returns

#<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the amplitude in dBm at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples

READ:OBWIDTH:SPECTRUM? might return #43204xxxx... (3204-byte data) for the spectrum trace data of the Occupied Bandwidth measurement.

READ:OBWidth:XDBBandwidth:BANDwidth? (Query Only)

Returns the x dB bandwidth in the Occupied Bandwidth measurement.

Conditions

Measurement views: Occupied Bandwidth

Group

Read commands

Syntax

`READ:OBWidth:XDBBandwidth:BANDwidth?`

Arguments

None

Returns

`<xdBBW>::<NRf>` is the x dB bandwidth in Hz.

Examples

`READ:OBWIDTH:XDBBANDWIDTH:BANDWIDTH?` might return `2.0E+6`, indicating the x dB bandwidth is 2 MHz.

READ:OBWidth:XDBBandwidth:LEFT:FREQuency? (Query Only)

Returns the left (lower) frequency of the x dB bandwidth.

Conditions

Measurement views: Occupied Bandwidth

Group

Read commands

Syntax

`READ:OBWidth:XDBBandwidth:LEFT:FREQuency?`

Related Commands

READ:OBWidth:XDBBandwidth:RIGHt:FREQuency? (on page 1694)

Arguments

None

Returns

`<xdBBW_left_freq>::<NRf>` is the left frequency in Hz.

Examples

`READ:OBWIDTH:XDBBANDWIDTH:LEFT:FREQUENCY?` might return `1.498E+9`, indicating the left frequency is 1.498 GHz.

READ:OBWidth:XDBBandwidth:LEFT:LEVel? (Query Only)

Returns the level at the left frequency of the x dB bandwidth.

Conditions

Measurement views: Occupied Bandwidth

Group

Read commands

Syntax

`READ:OBWidth:XDBBandwidth:LEFT:LEVel?`

Related Commands

READ:OBWidth:XDBBandwidth:RIGHt:LEVel? (on page 1695)

Arguments

None

Returns

`<xdBBW_left_level>::<NRf>` is the level at the left frequency in dB.

Examples

`READ:OBWIDTH:XDBBANDWIDTH:LEFT:LEVEL?` might return `-23.5`, indicating the level at the left frequency is -23.5 dB.

READ:OBWidth:XDBBandwidth:POWer? (Query Only)

Returns the reference power in the x dB bandwidth measurement.

Conditions

Measurement views: Occupied Bandwidth

Group

Read commands

Syntax

READ:OBWidth:XDBBandwidth:POWer?

Arguments

None

Returns

`<xdBBW_ref_power>::<NRf>` is the reference power in dBm. The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples

`READ:OBWIDTH:XDBBANDWIDTH:POWER?` might return `-10.0`, indicating the reference power is -10 dBm.

READ:OBWidth:XDBBandwidth:RIGHt:FREQuency? (Query Only)

Returns the right (higher) frequency of the x dB bandwidth.

Conditions

Measurement views: Occupied Bandwidth

Group

Read commands

Syntax

`READ:OBWidth:XDBBandwidth:RIGHt:FREQuency?`

Related Commands

READ:OBWidth:XDBBandwidth:LEFT:FREQuency? (on page 1693)

Arguments

None

Returns

`<xdBBW_right_freq>::<NRf>` is the right frequency in Hz.

Examples

`READ:OBWIDTH:XDBBANDWIDTH:RIGHT:FREQUENCY?` might return `1.502E+9`, indicating the right frequency is 1.502 GHz.

READ:OBWidth:XDBBandwidth:RIGHt:LEVel? (Query Only)

Returns the level at the right frequency of the x dB bandwidth.

Conditions

Measurement views: Occupied Bandwidth

Group

Read commands

Syntax

`READ:OBWidth:XDBBandwidth:RIGHt:LEVel?`

Related Commands

READ:OBWidth:XDBBandwidth:LEFT:LEVel? (on page 1693)

Arguments

None

Returns

`<xdBBW_right_level>::<NRf>` is the level at the right frequency in dB.

Examples

`READ:OBWIDTH:XDBBANDWIDTH:RIGHT:LEVEL?` might return `-23.5`, indicating the level at the right frequency is -23.5 dB.

READ:OFDM:APOWer? (Query Only)

Returns the average power in the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

`READ:OFDM:APOWer?`

Arguments

None

Returns

`<apower>::<NRf>` the average power in dB.

Examples

`READ:OFDM:APOWER?` might return `-23.4584459235` indicating the average power is −23.46 dB.

READ:OFDM:APOWer:PEAK? (Query Only)

Returns the peak-to-average power in the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

`READ:OFDM:APOWer:PEAK?`

Arguments

None

Returns

`<apower_peak>::<NRf>` is the peak-to-average power in dBm.

Examples

`READ:OFDM:APOWER:PEAK?` might return `10.4140096289`, indicating that peak-to-average power is 10.41 dBm.

READ:OFDM:CONSte:MAGNitude? (Query Only)

Returns the constellation magnitude data for the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

READ:OFDM:CONSte:MAGNitude?

Related Commands

READ:OFDM:CONSte:PHASe? (on page 1697)

READ:OFDM:CONSte:TYPE? (on page 1698)

READ:OFDM:CONSte:VALue? (on page 1699)

Arguments

None

Returns

#<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

READ:OFDM:CONSTE:MAGNITUDE? might return #510400xxxx... (10400-byte data) for the measurement.

READ:OFDM:CONSte:PHASe? (Query Only)

Returns the constellation phase data for the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

READ:OFDM:CONSte:PHASe?

Related Commands

READ:OFDM:CONSte:MAGNitude? (on page 1696)

READ:OFDM:CONSte:TYPE? (on page 1698)

READ:OFDM:CONSte:VALue? (on page 1699)

Arguments

None

Returns

#<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point in 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

READ:OFDM:CONSTE:PHASE? might return #510400xxxx... (10400-byte data) for the measurement.

READ:OFDM:CONSte:TYPE? (Query Only)

Returns the constellation context value of the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

READ:OFDM:CONSte:TYPE?

Related Commands

READ:OFDM:CONSte:MAGNitude? (on page 1696)

READ:OFDM:CONSte:PHASe? (on page 1697)

READ:OFDM:CONSte:VALue? (on page 1699)

Arguments

None

Returns

#<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point in 16-bit little endian integers. The integers must be translated to the proper context as listed below.

Integer value	Type (context)
0	Pilot
1	Data
2	Unused or Null

Examples

READ:OFDM:CONSTE:TYPE? might return #41352 (1352-byte data) for the measurement. The actual data must be decoded to the context type using the table above.

READ:OFDM:CONSte:VALue? (Query Only)

Returns the constellation value of the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

`READ:OFDM:CONSte:VALue?`

Related Commands

READ:OFDM:CONSte:MAGNitude? (on page 1696)

READ:OFDM:CONSte:PHASe? (on page 1697)

READ:OFDM:CONSte:TYPE? (on page 1698)

Arguments

None

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the nth data point in 16-bit little endian integers.

Examples

`READ:OFDM:CONSTE:VALUE?` might return `#41352`, indicating 1352 bytes of data for the measurement.

READ:OFDM:CPE? (Query Only)

Returns the Common Pilot Error magnitude for the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

`READ:OFDM:CPE?`

Arguments

None

Returns

<NRf> the RMS magnitude error in percent.

Examples

READ:OFDM:CPE? might return 3.7868041505, indicating that the CPE is 3.787%.

READ:OFDM:CRESponse:MAGNitude? (Query Only)

Returns the channel response magnitude data for the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

READ:OFDM:CRESponse:MAGNitude?

Arguments

None

Returns

#<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

READ:OFDM:CRESPONSE:MAGNITUDE? might return #3804xxxx. . . (804 byte data) for the measurement.

READ:OFDM:CRESponse:PHASe? (Query Only)

Returns the channel response phase data for the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

READ:OFDM:CRESponse:PHASe?

Arguments

None

Returns

#<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

READ:OFDM:CRESPONSE:PHASE? might return #3804xxxx. . . (804 byte data) for the measurement.

READ:OFDM:EVM:PEAK:DECibel:ALL? (Query Only)

Returns the peak EVM data for all subcarriers in the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

READ:OFDM:EVM:PEAK:DECibel:ALL?

Arguments

None

Returns

<NRf> data in dB.

Examples

READ:OFDM:EVM:PEAK:DECIBEL:ALL? might return -20.1872549032 indicating the data is −20.19 dB for the measurement.

READ:OFDM:EVM:PEAK:DECibel:DATA? (Query Only)

Returns the peak EVM data for the data subcarriers in the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

READ:OFDM:EVM:PEAK:DECibel:DATA?

Arguments

None

Returns

<NRf> data in dB.

Examples

READ:OFDM:EVM:PEAK:DECIBEL:DATA? might return -20.1872549032 indicating the data is -20.19 dB for the measurement.

READ:OFDM:EVM:PEAK:DECibel:PILots? (Query Only)

Returns the peak EVM data for the pilot subcarriers in the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

READ:OFDM:EVM:PEAK:DECibel:PILots?

Arguments

None

Returns

<NRf> data in dB.

Examples

READ:OFDM:EVM:PEAK:DECIBEL:PILOTS? might return -20.2872549032 indicating the data is -20.29 dB for the measurement.

READ:OFDM:EVM:PEAK:PERCent:ALL? (Query Only)

Returns the peak EVM data for all subcarriers in the OFDM measurement as a percent.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

READ:OFDM:EVM:PEAK:PERCent:ALL?

Arguments

None

Returns

<NRf> data in percent.

Examples

READ:OFDM:EVM:PEAK:PERCENT:ALL? might return 19.3223863840, indicating the data is 19.322% for the measurement.

READ:OFDM:EVM:PEAK:PERCent:DATA? (Query Only)

Returns the peak EVM data for the data subcarriers in the OFDM measurement as a percent.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

READ:OFDM:EVM:PEAK:PERCent:DATA?

Arguments

None

Returns

<NRf> data in percent.

Examples

READ:OFDM:EVM:PEAK:PERCENT:DATA? might return 19.3223863840 indicating the data is 19.322% for the measurement.

READ:OFDM:EVM:PEAK:PERCent:PILots? (Query Only)

Returns the peak EVM data for the pilot subcarriers in the OFDM measurement as a percent.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

READ:OFDM:EVM:PEAK:PERCent:PILots?

Arguments

None

Returns

<NRf> data in percent.

Examples

READ:OFDM:EVM:PEAK:PERCENT:PILOTS? might return 10.8543863840, indicating the data is 10.854% for the measurement.

READ:OFDM:EVM:PEAK:SCARrier:ALL? (Query Only)

Returns the peak EVM data for all subcarriers at the subcarrier level in the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

`READ:OFDM:EVM:PEAK:SCARrier:ALL?`

Arguments

None

Returns

`<NR1>`

Examples

`READ:OFDM:EVM:PEAK:SCARRIER:ALL?` might return `99.00000`, indicating the data is 99 for the measurement.

READ:OFDM:EVM:PEAK:SCARrier:DATA? (Query Only)

Returns the peak EVM data for the data subcarriers at the subcarrrier level in the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

`READ:OFDM:EVM:PEAK:SCARrier:DATA?`

Arguments

None

Returns

`<NR1>`

Examples

`READ:OFDM:EVM:PEAK:SCARRIER:DATA?` might return `99.00000`, indicating the data is 99 for the measurement.

READ:OFDM:EVM:PEAK:SCARrier:PILots? (Query Only)

Returns the peak EVM data for the pilot subcarriers at the subxcarrier level in the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

```
READ:OFDM:EVM:PEAK:SCARrier:PILots?
```

Arguments

None

Returns

<NR1>

Examples

READ:OFDM:EVM:PEAK:SCARRIER:PILOTS? might return 88.00000, indicating the data is 88 for the measurement.

READ:OFDM:EVM:PEAK:SYMBol:ALL? (Query Only)

Returns the peak EVM data for all subcarriers at the symbol level in the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

```
READ:OFDM:EVM:PEAK:SYMBol:ALL?
```

Arguments

None

Returns

<NR1>

Examples

READ:OFDM:EVM:PEAK:SYMBOL:ALL? might return 2.00000, indicating the data is 2 for the measurement

READ:OFDM:EVM:PEAK:SYMBol:DATA? (Query Only)

Returns the peak EVM data for the data subcarriers at the symbol level in the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

```
READ:OFDM:EVM:PEAK:SYMBol:DATA?
```

Arguments

<NR1>

Returns

Examples

READ:OFDM:EVM:PEAK:SYMBOL:DATA? might return 2.00000, indicating the data is 2 for the measurement

READ:OFDM:EVM:PEAK:SYMBol:PILots? (Query Only)

Returns the peak EVM data for the pilot subcarriers at the symbol level in the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

READ:OFDM:EVM:PEAK:SYMBol:PILots?

Arguments

<NR1>

Returns

Examples

READ:OFDM:EVM:PEAK:SYMBOL:PILOTS? might return 1.00000, indicating the data is 1 for the measurement

READ:OFDM:EVM:RMS:DECibel:ALL? (Query Only)

Returns the RMS EVM data for all subcarriers in the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

READ:OFDM:EVM:RMS:DECibel:ALL?

Arguments

None

Returns

<NRf> data in dB.

Examples

READ:OFDM:EVM:RMS:DECIBEL:ALL? might return -26.9012093267 indicating the data is -26.90 for the measurement.

READ:OFDM:EVM:RMS:DECibel:DATA? (Query Only)

Returns the RMS EVM data for the data subcarriers in the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

READ:OFDM:EVM:RMS:DECibel:DATA?

Arguments

None

Returns

<NRf> data in dB.

Examples

READ:OFDM:EVM:RMS:DECIBEL:DATA? might return -26.8477116269 indicating the data is -26.85 for the measurement.

READ:OFDM:EVM:RMS:DECibel:PILots? (Query Only)

Returns the RMS EVM data for the pilot subcarriers in the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

READ:OFDM:EVM:RMS:DECibel:PILots?

Arguments

None

Returns

<NRf> data in dB.

Examples

READ:OFDM:EVM:RMS:DECIBEL:PILOTS? might return -33.0589143032 indicating the data is -33.06 dB.

READ:OFDM:EVM:RMS:PERCent:ALL? (Query Only)

Returns the peak RMS data for all subcarriers in the OFDM measurement as a percent.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

`READ:OFDM:EVM:RMS:PERCent:ALL?`

Arguments

None

Returns

`<NRf>` data in percent.

Examples

`READ:OFDM:EVM:RMS:PERCENT:ALL?` might return `225.0743627548`, indicating the data is 225.074% for the measurement.

READ:OFDM:EVM:RMS:PERCent:DATA? (Query Only)

Returns the peak RMS data for the data subcarriers in the OFDM measurement as a percent.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

`READ:OFDM:EVM:RMS:PERCent:DATA?`

Arguments

None

Returns

`<NRf>` data in percent.

Examples

`READ:OFDM:EVM:RMS:PERCENT:DATA?` might return `260.6973409653`, indicating that the data is 260.697% for the measurement.

READ:OFDM:EVM:RMS:PERCent:PILots? (Query Only)

Returns the peak RMS data for the pilot subcarriers in the OFDM measurement as a percent.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

`READ:OFDM:EVM:RMS:PERCent:PILots?`

Arguments

None

Returns

`<NRf>` data in percent.

Examples

`READ:OFDM:EVM:RMS:PERCENT:PILOTS?` might return `210.8103863840`, indicating the data is 210.810% for the measurement.

READ:OFDM:EVM:TRACe<x>? (Query Only)

Returns the EVM trace data for the OFDM measurement.

When <x> is 1, the parameter is Matrix.

When <x> is 2, the parameter is Average versus Symbols.

When <x> is 3, the parameter is Average versus Subcarrier.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

`READ:OFDM:EVM:TRACe<x>?`

Arguments

None

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

READ:OFDM:EVM:TRACE2? might return #252xxxx. . . (52 byte data) for the Average versus Symbols measurement.

READ:OFDM:FERRor? (Query Only)

Returns the Frequency error reading for the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

READ:OFDM:FERRor?

Arguments

None

Returns

<NRf> frequency error.

Examples

READ:OFDM:FERROR? might return 82.8617142098E+3, indicating the frequency error was 82.86 kHz.

READ:OFDM:FLATness:PASS? (Query Only)

Queries whether the average power level values of subcarriers across the signal bandwidth on the OFDM Spectral Flatness display remain within the limits defined for a particular standard.

Conditions

Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group

Read commands

Syntax

READ:OFDM:FLATness:PASS?

Returns

Returns "1" in the case that all data remains within the defined limits, and "0" otherwise.

Examples

READ:OFDM:FLAT:PASS? might return 1, which means that the average power level values of subcarriers across the signal bandwidth remain within the defined limits.

READ:OFDM:FLATness:RESult? (Query Only)

Returns the summary results of each segment of the OFDM Spectral Flatness display.

Conditions

Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group

Read commands

Syntax

`READ:OFDM:FLATness:RESult?`

Returns

The data is returned as 12 comma delineated text values. The specific values are :

- minimum subcarrier within segment 1
- minimum subcarrier average energy deviation from the global average within segment 1
- deviation the of minimum subcarrier average from the allowed value in segment 1
- maximum subcarrier within segment 1
- maximum subcarrier average energy deviation from the global average within segment 1
- deviation of the maximum subcarrier average from the allowed value in segment 1
- minimum subcarrier within segment 2
- minimum subcarrier average energy deviation from the global average within segment 2
- deviation the of minimum subcarrier average from the allowed value in segment 2
- maximum subcarrier within segment 2
- maximum subcarrier average energy deviation from the global average within segment 2
- deviation of the maximum subcarrier average from the allowed value in segment 2

Examples

`READ:OFDM:FLAT:RES?` might return might return `-6.0000000000,-17.0626174659E-3,3.9829373825,12.0000000000,19.9792608`, which represents the summary results of each segment of the OFDM Spectral Flatness display.

READ:OFDM:FLATness:TRACe<x>? (Query Only)

Returns the OFDM Spectral Flatness trace data. When <x> is 1, the parameter is Matrix (symbol deviation per subcarrier). When<x> is 2, the parameter is Average Deviation vs Subcarrier.

Conditions

Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group

Read commands

Syntax

`READ:OFDM:FLATness:TRACe<x>?`

Returns

`#<num_digit><num_byte><data(1)><data(2)>..<data(n)>`

Where:

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

READ:OFDM:FLAT:TRACE2? might return #510400.. (10400-byte data) which represents the data for the Average Deviation vs. Subcarrier trace.

READ:OFDM:GAIN:IMBalance? (Query Only)

Returns the gain imbalance for the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

READ:OFDM:GAIN:IMBalance?

Arguments

None

Returns

<NRf> gain imbalance in dB.

Examples

READ:OFDM:GAIN:IMBALANCE? might return -57.746E-3 indicating that the gain imbalance is −0.057746 dB.

READ:OFDM:IQ:ORIGin:OFFSet? (Query Only)

Returns the IQ origin offset for the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

READ:OFDM:IQ:ORIGin:OFFSet?

Arguments

None

Returns

<NRf> gain imbalance in dB.

Examples

READ:OFDM:IQ:ORIGIN:OFFSET? might return −53.47017, indicating that the IQ origin offset is −53.47 dB.

READ:OFDM:MERRor:TRACe<x>? (Query Only)

Returns the magnitude error trace data for the OFDM measurement.

When <x> is 1, the parameter is Matrix.

When <x> is 2, the parameter is Average versus Symbols.

When <x> is 3, the parameter is Average versus Subcarrier.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

READ:OFDM:MERRor:TRACe<x>?

Arguments

None

Returns

#<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

READ:OFDM:MERROR:TRACE2 might return #3108xxxx. . . (108-byte data) for the average versus symbols trace measurement.

READ:OFDM:PACKet:DIRection? (Query Only)

Returns the direction of the packet in the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

READ:OFDM:PACKet:DIRection?

Arguments

None

Returns

One of the following values.

UP specifies that the packet is in the up direction.

DOWN specifies that the packet is in the down direction.

BOTH specifies that the packet is both directions.

Examples

`READ:OFDM:PACKET:DIRECTION?` might return `DOWN`, indicating that the packet was in the down direction.

READ:OFDM:PERRor:TRACe<x>? (Query Only)

Returns the phase error trace data for the OFDM measurement.

When <x> is 1, the parameter is Matrix.

When <x> is 2, the parameter is Average versus Symbols.

When <x> is 3, the parameter is Average versus Subcarrier.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

`READ:OFDM:PERRor:TRACe<x>?`

Arguments

None

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`READ:OFDM:PERROR:TRACE2` might return `#3100xxxx. . .` (100-byte data) for the average versus symbols trace measurement.

READ:OFDM:POWer:TRACe<x>? (Query Only)

Returns the power trace data for the OFDM measurement.

When <x> is 1, the parameter is Matrix.

When <x> is 2, the parameter is Average versus Symbols.

When <x> is 3, the parameter is Average versus Subcarrier.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

`READ:OFDM:POWer:TRACe<x>?`

Arguments

None

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`READ:OFDM:POWER:TRACE2` might return `#296xxxx. . .` (96-byte data) for the average versus symbols trace measurement.

READ:OFDM:QUADrature:OFFSet? (Query Only)

Returns the quadrature offset in the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

`READ:OFDM:QUADrature:OFFSet?`

Arguments

None

Returns

`<NRf>` quadrature offset in degrees.

Examples

READ:OFDM:QUADRATURE:OFFSET? might return -99.9, indicating that the offset is -99°.

READ:OFDM:SCARriers? (Query Only)

Returns the number of subcarriers in the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

READ:OFDM:SCARriers?

Arguments

None

Returns

<NR1> number of subcarriers.

Examples

READ:OFDM:SCARRIERS? might return 200.0000, indicating that there are 200 subcarriers in the measurement.

READ:OFDM:SCARriers:SPACing? (Query Only)

Returns the subcarrier spacing for the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

READ:OFDM:SCARriers:SPACing?

Arguments

None

Returns

<NRf> subcarrier spacing.

Examples

READ:OFDM:SCARRIERS:SPACING? might return 90.000E+3, indicating that the subcarrier spacing is 90.000 kHz.

READ:OFDM:STABle:VALUe? (Query Only)

Returns the number of bytes in the symbol table for the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

`READ:OFDM:STABle:VALUe?`

Arguments

None

Returns

`#<num_digit><num_byte>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

Examples

`READ:OFDM:STABLE:VALUE?` might return `#43848`, indicating that there are 3848 bytes in the symbol table.

READ:OFDM:SYMBol:CERRor? (Query Only)

Returns the symbol clock error for the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

`READ:OFDM:SYMBol:CERRor?`

Arguments

None

Returns

`<NRf>` symbol clock error in ppm.

Examples

`READ:OFDM:SYMBOL:CERROR?` might return `422.7135479929`, indicating that the symbol clock error was 422.714 ppm.

READ:OFDM:SYMBol:COUNt? (Query Only)

Returns the number of symbols for the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Read commands

Syntax

`READ:OFDM:SYMBol:COUNt?`

Arguments

None

Returns

`<NR1>` number of symbols in the measurement.

Examples

`READ:OFDM:SYMBOL:COUNT?` might return `25.00000`, indicating that there were 25 symbols for the measurement.

READ:P25:CONSte:FERRor? (Query Only)

Returns the frequency error in Hz. The frequency error is the difference between the measured carrier frequency of the signal and the user-selected center frequency of the analyzer.

When Frequency Error is in Auto mode, this query returns the frequency error value. When in Manual mode, it returns the frequency offset.

Conditions

Measurement view: P25 Constellation

Group

Read commands

Syntax

`READ:P25:CONSte:FERRor?`

Returns

`<freq_error>::<NRf>` is the frequency error in Hz.

Examples

`READ:P25:CONSTE:FERROR?` might return `-10.7E+3`, which is a frequency error of -10.7 kHz.

READ:P25:EDIagram:FDEViation? (Query Only)

Returns the Frequency Deviation vs Time trace data with the X values.

Conditions

Measurement view: P25 Eye Diagram

Group

Read commands

Syntax

READ:P25:EDIagram:FDEViation?

Returns

#<num_digit><num_byte><Y(1)><X(1)><Y(2)><X(2)>...<Y(n)><X(n)>

Where:

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<Y(n)> and <X(n)> are the frequency deviation in Hz and time (symbols) coordinate pair at the nth data point. 4-byte little endian floating point format specified in IEEE 488.2.

Examples

READ:P25:EDIAGRAM:FDEVIATION? might return #3160xxxx... (160-byte data) for the Frequency Deviation vs Time trace.

READ:P25:EDIagram:FERRor? (Query Only)

Returns the frequency error in Hz. The frequency error is the difference between the measured carrier frequency of the signal and the user-selected center frequency of the analyzer.

Conditions

Measurement view: P25 Eye Diagram

Group

Read commands

Syntax

READ:P25:EDIagram:FERRor?

Returns

<freq_error>::<NRf> is the frequency error in Hz.

Examples

READ:P25:EDIAGRAM:FERROR? might return -10.7E+3, which is a frequency error of -10.7 kHz.

READ:P25:PVTime:BURSt:POWer? (Query Only)

Returns the average power of the burst packet, in dBm, for the P25 Power vs. Time trace.

Conditions

Measurement view: P25 Power vs. Time

Group

Read commands

Syntax

READ:P25:PVTime:BURSt:POWer?

Returns

Floating point number that represents the average power of the burst packet, in dBm.

Examples

`READ:P25:PVTIME:BURST:POWER?` might return `-1.99`, which represents the value of the average power of the burst packet of -1.99 dBm.

READ:P25:PVTime:FALL:TIMe? (Query Only)

Queries the value of the width of the falling edge, in seconds.

Conditions

Measurement view: P25 Power vs. Time

Group

Read commands

Syntax

`READ:P25:PVTime:FALL:TIMe?`

Related Commands

READ:P25:PVTime:FTIMe? (on page 1720)

Returns

Floating point number which represents the value of the width of the falling edge of the burst, in seconds.

Examples

`READ:P25:PVTIME:FALL:TIME?` might return `7e-6`, which indicates the width of the falling edge is 7 µs.

READ:P25:PVTime:FTIMe? (Query Only)

Queries the value of the width of the falling edge, in seconds.

Conditions

Measurement view: P25 Power vs. Time

Group

Read commands

Syntax

`READ:P25:PVTime:FTIMe?`

Related Commands

READ:P25:PVTime:FALL:TIMe? (on page 1720)

Returns

Floating point number which represents the value of the width of the falling edge of the burst, in seconds.

Examples

`READ:P25:PVTIME:FTIME?` might return `7e-6`, which indicates the width of the falling edge is 7 µs.

READ:P25:PVTime:TRACe:X? (Query Only)

Returns the horizontal values (time in seconds) for the P25 Power vs. Time trace.

Conditions

Measurement view: P25 Power vs. Time

Group

Read commands

Syntax

`READ:P25:PVTime:TRACe:X?`

Returns

`#<num_digit><num_byte><data(1)><data(2)>..<data(n)>`

Where:

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`READ:P25:PVTIME:TRACE:X?` might return `#43204` (3204-byte data), which represents the value of the time in seconds (horizontal values) for the P25 Power vs. Time trace.

READ:P25:PVTime:TRACe:XY? (Query Only)

Returns the horizontal value (time in seconds) and vertical value (power) for the P25 Power vs. Time trace <x(n)><y(n)> is the horizontal value and vertical value pair at the nth data point.

Conditions

Measurement view: P25 Power vs. Time

Group

Read commands

Syntax

`READ:P25:PVTime:TRACe:XY?`

Related Commands

READ:P25:PVTime:TRACe:X? (on page 1721)

Returns

`#<num_digit><num_byte><data(1)><data(2)>..<data(n)>`

Where:

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`READ:P25:PVTIME:TRACE:XY?` might return #43204 (3204-byte data), which represents the horizontal value (time in seconds) and vertical value (power) pair at the nth data point.

READ:P25:SUMMary:LIMIt:SRA (Query Only)

Returns Limit of Symbol Rate Accuracy measurement results.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

`READ:P25:SUMMary:LIMIt:SRA`

Arguments

None

Returns

might return PASS, FAIL or NA.

Examples

`READ:P25:SUMMARY:LIMIT:SRA`

READ:P25:SUMMary:MODUlation:MEASurement:FIDelity? (Query Only)

Returns the Modulation Fidelity measurement results.

Conditions

Measurement view: P25 Summary

Group

Read commands

Syntax

`READ:P25:SUMMary:MODUlation:MEASurement:FIDelity?`

Arguments

None

Returns

Modulation Fidelity measurement result

Examples

`READ:P25:SUMMARY:MODULATION:MEASUREMENT:FIDELITY?`

READ:P25:SUMMary:MODUlation:MEASurement:FIDelity:STATus (Query Only)

Returns the Modulation Fidelity measurement status.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

`READ:P25:SUMMary:MODUlation:MEASurement:FIDelity:STATus?`

Arguments

None

Examples

`READ:P25:SUMMARY:MODULATION:MEASUREMENT:FIDELITY:STATUS?` might return `PASS`, `FAIL`, or `NA`.

READ:P25:SUMMary:MODUlation:MEASurement:FREQuency:ACCUracy (No Query Form)

Returns the Frequency Accuracy measurement's result.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

`READ:P25:SUMMary:MODUlation:MEASurement:FREQuency:ACCUracy`

Arguments

None

Examples

`READ:P25:SUMMARY:MODULATION:MEASUREMENT:FREQUENCY:ACCURACY`

READ:P25:SUMMary:MODUlation:MEASurement:FREQuency:ACCUracy? (Query Only)

Returns the Operational Frequency Accuracy measurement results.

Conditions

Measurement view: P25 Summary

Group

Read commands

Syntax

```
READ:P25:SUMMary:MODUlation:MEASurement:FREQuency:ACCUracy?
```

Arguments

None

Returns

Frequency Accuracy measurement result

Examples

```
READ:P25:SUMMARY:MODULATION:MEASUREMENT:FREQUENCY:ACCURACY?
```

READ:P25:SUMMary:MODUlation:MEASurement: FREQuency: ACCUracy:STATus (Query Only)

Returns the status of Frequency Accuracy measurement status.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

```
READ:P25:SUMMary:MODUlation:MEASurement:FREQuency:ACCUracy:STATus?
```

Arguments

None

Examples

`READ:P25:SUMMARY:MODULATION:MEASUREMENT:FREQUENCY:ACCURACY:STATUS?` might return `PASS`, `FAIL`, or `NA`.

READ:P25:SUMMary:MODUlation:MEASurement:FREQuency: DEViation:NEGPeak

Returns the Frequency Deviation and negative peak measurement results.

Conditions

Measurement view: P25 Summary

Group

Read commands

Syntax

READ:P25:SUMMary:MODUlation:MEASurement:FREQuency:DEVIation:NEGPeak

Arguments

<integer,integer>

Returns

Frequency Deviation, negative peak, result

Examples

READ:P25:SUMMARY:MODULATION:MEASUREMENT:FREQUENCY:DEVIATION:NEGPEAK <2,6> indicates the coordinate value of the Frequency Deviation negative peak.

READ:P25:SUMMary:MODUlation:MEASurement:FREQuency:DEVIation:NEGPeak:STATus

Returns the Status of Frequency Deviation, negative peak status.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

READ:P25:SUMMary:MODUlation:MEASurement:FREQuency:DEVIation:NEGPeak:STATus?

Arguments

<Integer,Integer>

Examples

READ:P25:SUMMARY:MODULATION:MEASUREMENT:FREQUENCY:DEVIATION:NEGPEAK:STATUS? might return PASS, FAIL, or NA.

READ:P25:SUMMary:MODUlation:MEASurement:FREQuency:DEVIation:POSPeak

Returns the Frequency Deviation, positive peak, measurement results.

Conditions

Measurement view: P25 Summary

Group

Read commands

Syntax

READ:P25:SUMMary:MODUlation:MEASurement:FREQuency:DEVIation:POSPeak

Arguments

<integer,integer>

Returns

Frequency Deviation, positive peak, result

Examples

READ:P25:SUMMARY:MODULATION:MEASUREMENT:FREQUENCY:DEVIATION:POSPEAK <-2,-6> indicates the coordinate value of the Frequency Deviation positive peak.

READ:P25:SUMMary:MODUlation:MEASurement:FREQuency:DEVIation:POSPeak:STATus

Returns the status of Frequency Deviation, positive peak status.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

READ:P25:SUMMary:MODUlation:MEASurement:FREQuency:DEVIation:POSPeak:STATus?

Arguments

None

Examples

READ:P25:SUMMARY:MODULATION:MEASUREMENT:FREQUENCY:DEVIATION:POSPEAK:STATUS? might return PASS, FAIL, or NA.

READ:P25:SUMMary:MODUlation:MEASurement:FREQuency:ERRor

Returns the Frequency Error measurement results.

Conditions

Measurement view: P25 Summary

Group

Read commands

Syntax

READ:P25:SUMMary:MODUlation:MEASurement:FREQuency:ERRor?

Arguments

None

Returns

Frequency Error measurement result

Examples

```
READ:P25:SUMMARY:MODULATION:MEASUREMENT:FREQUENCY:ERROR?
```

READ:P25:SUMMary:MODUlation:MEASurement:FREQuency:ERRor (Query Only)

Returns the Frequency Error measurement's result.

Conditions

Measurement views: P25 Summary

Group

Read commands

Syntax

```
READ:P25:SUMMary:MODUlation:MEASurement:FREQuency:ERRor
```

Arguments

None

Examples

```
READ:P25:SUMMARY:MODULATION:MEASUREMENT:FREQUENCY:ERROR
```

READ:P25:SUMMary:MODUlation:MEASurement:SRA

Returns the Symbol Rate Accuracy measurement results.

Conditions

Measurement view: P25 Summary

Group

Read commands

Syntax

```
READ:P25:SUMMary:MODUlation:MEASurement:SRA
```

Arguments

None

Returns

Symbol Rate Accuracy measurement result

Examples

```
READ:P25:SUMMARY:MODULATION:MEASUREMENT:SRA?
```

READ:P25:SUMMary:MODUlation:MEASurement:SRA:STATus (Query Only)

Returns Status of Symbol Rate Accuracy measurement status.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

```
READ:P25:SUMMary:MODUlation:MEASurement:SRA:STATus
```

Arguments

None

Returns

Might return PASS, FAIL or NA.

Examples

```
READ:P25:SUMMARY:MODULATION:MEASUREMENT:SRA:STATUS
```

READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXOFF (Query Only)

Returns HCPM Pmax-off measurement results.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

```
READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXOFF
```

Arguments

None

Examples

```
READ:P25:SUMMARY:POWER:MEASUREMENT:HCPM:MAXOFF
```

READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXOFF:STATus? (Query Only)

Returns HCPM Pmax-off measurement status.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

```
READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXOFF:STATus?
```

Arguments

None

Examples

`READ:P25:SUMMARY:POWER:MEASUREMENT:HCPM:MAXOFF:STATUS?` might return `PASS`, `FAIL`, or `NA`.

READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXOFF? (Query Only)

This command returns the HCPM Pmax-off measurement results.

Conditions

Measurement view: P25 Summary

Group

Read commands

Syntax

```
READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXOFF?
```

Returns

HCPM Pmax-off measurement results

Examples

```
READ:P25:SUMMARY:POWER:MEASUREMENT:HCPM:MAXOFF?
```

READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXON (Query Only)

Returns HCPM Pmax-on measurement results.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

```
READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXON
```

Arguments

None

Examples

```
READ:P25:SUMMARY:POWER:MEASUREMENT:HCPM:MAXON
```

READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXON:STATus (Query Only)

Returns Status of HCPM Pmax-on measurement status.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

```
READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXON:STATus
```

Arguments

None

Examples

`READ:P25:SUMMARY:POWER:MEASUREMENT:HCPM:MAXON:STATUS` might return `PASS`, `FAIL`, or `NA`.

READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXON? (Query Only)

This command returns the HCPM Pmax-on measurement results.

Conditions

Measurement view: P25 Summary

Group

Read commands

Syntax

```
READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXON?
```

Returns

HCPM Pmax-on measurement results

Examples

```
READ:P25:SUMMARY:POWER:MEASUREMENT:HCPM:MAXON?
```

READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXSs (Query Only)

Returns HCPM Pss-max measurement results.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

`READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXSs`

Arguments

None

Examples

`READ:P25:SUMMARY:POWER:MEASUREMENT:HCPM:MAXSS`

READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXSs:STATus (Query Only)

Returns HCPM Pss-max measurement status.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

`READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXSs:STATus?`

Arguments

None

Examples

`READ:P25:SUMMARY:POWER:MEASUREMENT:HCPM:MAXSS:STATUS?` might return `PASS`, `FAIL`, or `NA`.

READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXSs? (Query Only)

Returns the HCPM Pss-max measurement results.

Conditions

Measurement view: P25 Summary

Group

Read commands

Syntax

```
READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXSs?
```

Returns

HCPM Pss-on measurement results

Examples

```
READ:P25:SUMMARY:POWER:MEASUREMENT:HCPM:MAXSS?
```

READ:P25:SUMMary:POWEr:MEASurement:HCPM:MINImum (Query Only)

Returns HCPM Pss-min measurement results.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

```
READ:P25:SUMMary:POWEr:MEASurement:HCPM:MINImum
```

Arguments

None

Examples

```
READ:P25:SUMMARY:POWER:MEASUREMENT:HCPM:MINIMUM
```

READ:P25:SUMMary:POWEr:MEASurement:HCPM:MINImum:STATus? (Query Only)

Returns Status of HCPM Pss-min measurement results based on the limit.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

```
READ:P25:SUMMary:POWEr:MEASurement:HCPM:MINImum:STATus?
```

Arguments

None

Examples

`READ:P25:SUMMARY:POWER:MEASUREMENT:HCPM:MINIMUM:STATUS?` might return `PASS`, `FAIL`, or `NA`.

READ:P25:SUMMary:POWEr:MEASurement:HCPM:MINImum? (Query Only)

Returns the HCPM Pss-min measurement results.

Conditions

Measurement view: P25 Summary

Group

Read commands

Syntax

`READ:P25:SUMMary:POWEr:MEASurement:HCPM:MINImum?`

Returns

HCPM Pss-min measurement results

Examples

`READ:P25:SUMMARY:POWER:MEASUREMENT:HCPM:MINIMUM?`

READ:P25:SUMMary:POWEr:MEASurement:HCPM:OFFSlot (Query Only)

Returns HCPM Offslot measurement results.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements

Group

Read commands

Syntax

`READ:P25:SUMMary:POWEr:MEASurement:HCPM:OFFSlot`

Arguments

None

Examples

`READ:P25:SUMMARY:POWER:MEASUREMENT:HCPM:OFFSLOT`

READ:P25:SUMMary:POWEr:MEASurement:HCPM:OFFSlot:STATus? (Query Only)

Returns Status of HCPM Offslot measurement based on the limit.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

```
READ:P25:SUMMary:POWEr:MEASurement:HCPM:OFFSlot:STATus?
```

Arguments

None

Examples

`READ:P25:SUMMARY:POWER:MEASUREMENT:HCPM:OFFSLOT:STATUS?` might return `PASS`, `FAIL`, or `NA`.

READ:P25:SUMMary:POWEr:MEASurement:HCPM:PEAK:ACPR:HI (Query Only)

Returns P_ACP_HI value which is displayed on P25 Summary display.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

```
READ:P25:SUMMary:POWEr:MEASurement:HCPM:PEAK:ACPR:HI?
```

Arguments

None

Examples

`READ:P25:SUMMARY:POWER:MEASUREMENT:HCPM:PEAK:ACPR:HI?` might return `-37.963`, indicating P_ACP_HI value is -37.96 dBm.

READ:P25:SUMMary:POWEr:MEASurement:HCPM:PEAK:ACPR:LOW (Query Only)

Returns P_ACP_LOW value which is displayed on P25 Summary display.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

`READ:P25:SUMMary:POWEr:MEASurement:HCPM:PEAK:ACPR:LOW?`

Arguments

None

Examples

`READ:P25:SUMMARY:POWER:MEASUREMENT:HCPM:PEAK:ACPR:LOW?` might return `-40.125`, indicating P_ACP_LOW value is –40.13 dBm.

READ:P25:SUMMary:POWEr:MEASurement:HCPM:PEAK:ACPR:MIN (Query Only)

Returns Min Pk ACPR value which is displayed on P25 Summary Display.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

`READ:P25:SUMMary:POWEr:MEASurement:HCPM:PEAK:ACPR:MIN?`

Arguments

None

Examples

`READ:P25:SUMMARY:POWER:MEASUREMENT:HCPM:PEAK:ACPR:MIN?` might return `47.96`, indicating Min Pk ACPR value is 47.96 dBm

READ:P25:SUMMary:POWEr:MEASurement:HCPM:PEAK:ACPR:MIN:STATus(Query Only)

Returns Status of Min Pk ACPR value based on the limit.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

`READ:P25:SUMMary:POWEr:MEASurement:HCPM:PEAK:ACPR:MIN:STATus`

Arguments

None

Examples

`READ:P25:SUMMARY:POWER:MEASUREMENT:HCPM:PEAK:ACPR:MIN:STATUS` might return PASS, FAIL or NA.

READ:P25:SUMMary:POWEr:MEASurement:RF:STATus (Query Only)

Returns status of RF measurement results.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

`READ:P25:SUMMary:POWEr:MEASurement:RF:STATus?`

Arguments

None

Examples

`READ:P25:SUMMARY:POWER:MEASUREMENT:RF:STATUS?` might return `PASS`, `FAIL`, or `NA`.

READ:P25:SUMMary:POWEr:MEASurement:RF (Query Only)

Returns RF measurement results.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

`READ:P25:SUMMary:POWEr:MEASurement:RF`

Arguments

None

Examples

`READ:P25:SUMMARY:POWER:MEASUREMENT:RF`

READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ONE (Query Only)

Returns HCPM Time alignment's t_error_1 measurement results.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

```
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ONE
```

Arguments

None

Examples

```
READ:P25:SUMMARY:TRIGGER:MEASUREMENT:HCPM:TIME:ALIGNMENT:TERROR:ONE
```

READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ONE:ACQ:COUNT? (Query Only)

Returns the HCPM Time alignment's t_error_1 acquisition count over which the t_error_1 averaging is done.

Conditions

Measurement view: P25 Summary

Group

Read commands

Syntax

```
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ONE:ACQ:COUNT?
```

Returns

HCPM Time alignment's t_error_1 acquisition count over which the t_error_1 averaging is done.

Examples

```
READ:P25:SUMMARY:TRIGGER:MEASUREMENT:HCPM:TIME:ALIGNMENT:TERROR:ONE:ACQ:COUNT?
```

READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ONE:STATus (Query Only)

Returns status of HCPM Time alignment's t_error_1 measurement results based on the limit.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

```
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ONE:STATus?
```

Arguments

None

Examples

`READ:P25:SUMMARY:TRIGGER:MEASUREMENT:HCPM:TIME:ALIGNMENT:TERROR:ONE:STATUS?` might return `PASS`, `FAIL`, or `NA`.

READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ONE?(Query Only)

Returns the HCPM Time alignment t_error_1 measurement results.

Conditions

Measurement view: P25 Summary

Group

Read commands

Syntax

`READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ONE?`

Returns

HCPM Time alignment t_error_1 measurement results.

Examples

`READ:P25:SUMMARY:TRIGGER:MEASUREMENT:HCPM:TIME:ALIGNMENT:TERROR:ONE?`

READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ZERO (Query Only)

Returns HCPM Time alignment's t_error_0 measurement results.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

`READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ZERO?`

Arguments

None

Examples

`READ:P25:SUMMARY:TRIGGER:MEASUREMENT:HCPM:TIME:ALIGNMENT:TERROR:ZERO?`

READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ZERO:ACQ:COUNT? (Query Only)

Returns the HCPM Time alignment's t_error_0 acquisition count over which the t_error_0 averaging is done.

Conditions

Measurement view: P25 Summary

Group

Read commands

Syntax

`READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ZERO:ACQ:COUNT?`

Returns

HCPM Time alignment's t_error_0 acquisition count.

Examples

`READ:P25:SUMMARY:TRIGGER:MEASUREMENT:HCPM:TIME:ALIGNMENT:TERROR:ZERO:ACQ:COUNT?`

READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ZERO:STATus (Query Only)

Returns status of HCPM Time alignment's t_error_0 measurement results.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

`READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ZERO:STATus?`

Arguments

None

Examples

`READ:P25:SUMMARY:TRIGGER:MEASUREMENT:HCPM:TIME:ALIGNMENT:TERROR:ZERO:STATUS?` might return PASS, FAIL or NA.

READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ZERO? (Query Only)

Returns the HCPM Time alignment's t_error_0 measurement results over which the t_error_0 averaging is done.

Conditions

Measurement view: P25 Summary

Group

Read commands

Syntax

```
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ZERO?
```

Returns

HCPM Time alignment's t_error_0 measurement results over which the t_error_0 averaging is done.

Examples

```
READ:P25:SUMMARY:TRIGGER:MEASUREMENT:HCPM:TIME:ALIGNMENT:TERROR:ZERO?
```

READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TOBSync (Query Only)

Returns HCPM Time alignment's t_obsync measurement results.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

```
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TOBSync?
```

Arguments

None

Examples

```
READ:P25:SUMMARY:TRIGGER:MEASUREMENT:HCPM:TIME:ALIGNMENT:TOBSYNC?
```

READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ACQ:COUNT (Query Only)

Returns Average Transmitter Power Attack acquisition count.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

```
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ACQ:COUNT?
```

Arguments

None

Examples

```
READ:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:ACQ:COUNT?
```

READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TOBSync? (Query Only)

Returns the HCPM Time alignment's t_obsync measurement results.

Conditions

Measurement view: P25 Summary

Group

Read commands

Syntax

```
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TOBSync?
```

Returns

HCPM Time alignment's t_obsync measurement results.

Examples

```
READ:P25:SUMMARY:TRIGGER:MEASUREMENT:HCPM:TIME:ALIGNMENT:TOBSYNC?
```

READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ACQ:COUNT (Query Only)

Returns Average Transmitter Power Attack acquisition count.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

```
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ACQ:COUNT?
```

Arguments

None

Examples

```
READ:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:ACQ:COUNT?
```

READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ACQ:COUNT:BI (Query Only)

Returns Average Transmitter Power Attack, busy/idle measurement results.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

```
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ACQ:COUNT:BI?
```

Arguments

None

Examples

```
READ:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:ACQ:COUNT:BI?
```

READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TOBSync:ACQ:COUNT (Query Only)

Returns HCPM Time alignment's t_obsync acquisition counts.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

```
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TOBSync:ACQ:COUNT?
```

Arguments

None

Examples

```
READ:P25:SUMMARY:TRIGGER:MEASUREMENT:HCPM:TIME:ALIGNMENT:TOBSYNC:ACQ:COUNT?
```

READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ENCOder:BI:STATus (Query Only)

Returns status of Average Transmitter Power Encoder Attack Time measurement results.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

```
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ENCOder:BI:STATus?
```

Arguments

None

Examples

`READ:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:ENCODER:BI:STATUS?` might return `PASS`, `FAIL`, or `NA`.

READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ACQ:COUNT:BI?(Query Only)

Returns the Average Transmitter Power Attack time busy/idle acquisition count over which the Phase 1 Attack time averaging is done.

Conditions

Measurement view: P25 Summary

Group

Read commands

Syntax

```
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ACQ:COUNT:BI?
```

Returns

Average Transmitter Power Attack, busy/idle, measurement results.

Examples

```
READ:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:ACQ:COUNT:BI?
```

READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ENCOder:STATus (Query Only)

Returns Status of Average Transmitter Encoder Attack Time measurement results based on the limit.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

```
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ENCOder:STATus?
```

Arguments

None

Examples

`READ:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:ENCODER:STATUS?` might return `PASS`, `FAIL`, or `NA`.

READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ENCOder? (Query Only)

Returns the Average Transmitter Encoder Attack Time measurement results

Conditions

Measurement view: P25 Summary

Group

Read commands

Syntax

`READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ENCOder?`

Returns

Average Transmitter Encoder Attack Time measurement results.

Examples

`READ:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:ENCODER?`

READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr (Query Only)

Returns Average Transmitter Power Attack Time measurement results.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

`READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr?`

Arguments

None

Examples

`READ:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:POWER?`

READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr:BI (Query Only)

Returns Average Transmitter Power Attack Time, busy/idle measurement results.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

`READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr:BI?`

Arguments

None

Examples

`READ:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:POWER:BI?`

READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr:BI:STATus (Query Only)

Returns Status of Average Transmitter Power Attack Time, busy/idle measurement results based on the Limit.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

`READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr:BI:STATus?`

Arguments

None

Examples

`READ:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:POWER:BI:STATUS?` might return PASS, FAIL or NA.

READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr:BI? (Query Only)

Returns the Average Transmitter Power Attack Time, busy/idle measurement results.

Conditions

Measurement view: P25 Summary

Group

Read commands

Syntax

```
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr:BI?
```

Returns

Average Transmitter Power Attack Time, busy/idle, measurement results.

Examples

```
READ:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:POWER:BI?
```

READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr:STATus (Query Only)

Returns Status of Average Transmitter Power Attack Time measurement results.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

```
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr:STATus?
```

Arguments

None

Examples

`READ:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:POWER:STATUS?` might return PASS, FAIL or NA.

READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr? (Query Only)

Returns the Average Transmitter Power Attack Time measurement results.

Conditions

Measurement view: P25 Summary

Group

Read commands

Syntax

```
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr?
```

Returns

Average Transmitter Power Attack Time measurement results.

Examples

```
READ:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:POWER?
```

READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy (Query Only)

Returns Throughput delay measurement results.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

```
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy?
```

Arguments

None

Examples

```
READ:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:THRUPUT:DELAY?
```

READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy:ACQ:COUNT (Query Only)

Returns Throughput delay acquisition counts.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

```
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy:ACQ:COUNT?
```

Arguments

None

Examples

```
READ:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:THRUPUT:DELAY:ACQ:COUNT?
```

READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy:ACQ:COUNT? (Query Only)

Returns the Throughput delay acquisition counts over which the Throughput delay averaging is done.

Conditions

Measurement view: P25 Summary

Group

Read commands

Syntax

`READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy:ACQ:COUNT?`

Returns

Throughput delay acquisition counts

Examples

`READ:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:THRUPUT:DELAY:ACQ:COUNT?`

READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy:STATus (Query Only)

Returns Status of Throughput delay measurement results based on the limit.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Read commands

Syntax

`READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy:STATus?`

Arguments

None

Examples

`READ:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:THRUPUT:DELAY:STATUS?` might return `PASS`, `FAIL`, or `NA`.

READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy? (Query Only)

Returns the Throughput delay measurement results.

Conditions

Measurement view: P25 Summary

Group

Read commands

Syntax

`READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy?`

Returns

Throughput delay measurement results.

Examples

`READ:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:THRUPUT:DELAY?`

READ:PERRor[:MEASview<y>]:FERRor? (Query only)

Returns the frequency error in the Phase error versus Time measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Phase error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Phase error versus Time).

Group

Read commands

Syntax

`READ:PERRor[:MEASview<y>]:FERRor?`

Arguments

None

Returns

`<freq_error>::<NRf>` is the frequency error in Hz.

Examples

`READ:PERRor:MEASview2:FERRor?` might return -10.7E+3, indicating the frequency error is -10.7 kHz for measurement view 2.

READ:PERRor[:MEASview<y>]:PEAK? (Query only)

Returns the peak value in the Phase error versus Time measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Phase error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Phase error versus Time).

Group

Read commands

Syntax

`READ:PERRor[:MEASview<y>]:PEAK?`

Related Commands

READ:PERRor[:MEASview<y>]:PINDex? (on page 1750)

Arguments

None

Returns

`<peak>::<NRf>` is the peak phase error in percent (%).

Examples

`READ:PERRor:MEASview2:PEAK?` might return 1.57, indicating the peak phase error value is 1.57% for measurement view 2.

READ:PERRor[:MEASview<y>]:PINDex? (Query only)

Returns the time at the phase error peak.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Phase error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Phase error versus Time).

Group

Read commands

Syntax

`READ:PERRor[:MEASview<y>]:PINDex?`

Related Commands

READ:PERRor[:MEASview<y>]:PEAK? (on page 1749)

Arguments

None

Returns

`<peak_time>::<NRf>` is the time at the phase error peak in symbol number. The unit can be changed by `[SENSe]:DDEMod:TIMe:UNITs` command.

Examples

`READ:PERRor:MEASview2:PINDex?` might return 68.000, indicating that the phase error peak is at symbol #68 for measurement view 2.

READ:PERRor[:MEASview<y>]:RMS (Query only)

Returns the RMS (Root-Mean-Square) value in the Phase error versus Time measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Phase error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Phase error versus Time).

Group

Read commands

Syntax

`READ:PERRor[:MEASview<y>]:RMS?`

Arguments

None

Returns

`<rms>::<NRf>` is the RMS phase error in percent (%).

Examples

`READ:PERRor:MEASview2:RMS?` might return 0.382, indicating the RMS PERRor value is 0.382% for measurement view 2.

READ:PERRor[:MEASview<y>]:TRACe? (Query only)

Returns the Phase error versus Time trace data.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Phase error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Phase error versus Time).

Group

Read commands

Syntax

`READ:PERRor[:MEASview<y>]:TRACe?`

Arguments

None

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the phase error versus Time trace data for the point n in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

READ:PERRor:MEASview2:TRACe? might return #42036xxxx... (2036-byte data) for the Phase error versus Time measview 2.

READ:PHVTime[:MEASview<y>]? (Query Only)

Returns the Phase versus Time trace data.

Conditions

Measurement views: Phase versus Time

Group

Read commands

Syntax

READ:PHVTime[:MEASview<y>]?

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

#<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the phase in degrees at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

READ:PHVTIME:MEASview1? might return #3160xxxx... (160-byte data) for the Phase versus Time trace.

READ:PHVTime[:MEASview<y>]:MAXimum? (Query Only)

Returns the maximum value in the Phase versus Time measurement.

Conditions

Measurement views: Phase versus Time

Group

Read commands

Syntax

```
READ:PHVTime[:MEASview<y>]:MAXimum?
```

Related Commands

READ:PHVTime[:MEASview<y>]:MAXLocation? (on page 1753)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

<max>::<NRf> is the maximum phase in degrees.

Examples

READ:PHVTIME:MEASview1:MAXIMUM? might return 153.8, indicating the maximum phase is 153.8 °.

READ:PHVTime[:MEASview<y>]:MAXLocation? (Query Only)

Returns the time at which the phase is maximum.

Conditions

Measurement views: Phase versus Time

Group

Read commands

Syntax

```
READ:PHVTime[:MEASview<y>]:MAXLocation?
```

Related Commands

READ:PHVTime[:MEASview<y>]:MAXimum? (on page 1752)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

<max_time>::<NRf> is the time in seconds at which the phase is maximum.

Examples

READ:PHVTIME:MEASview1:MAXLOCATION? might return 175.3E-9, indicating the I or Q level is maximum at 175.3 ns.

READ:PHVTime[:MEASview<y>]:MINimum? (Query Only)

Returns the minimum value in the Phase versus Time measurement.

Conditions

Measurement views: Phase versus Time

Group

Read commands

Syntax

`READ:PHVTime[:MEASview<y>]:MINimum?`

Related Commands

READ:PHVTime[:MEASview<y>]:MINLocation? (on page 1754)

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

`<min>::<NRf>` is the minimum phase in degrees.

Examples

`READ:PHVTIME:MEASview1:MINIMUM?` might return `-176.3`, indicating the minimum phase is −176.3°.

READ:PHVTime[:MEASview<y>]:MINLocation? (Query Only)

Returns the time at which the phase is minimum.

Conditions

Measurement views: Phase versus Time

Group

Read commands

Syntax

`READ:PHVTime[:MEASview<y>]:MINLocation?`

Related Commands

READ:PHVTime[:MEASview<y>]:MINimum? (on page 1753)

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

`<min_time>::<NRf>` is the time in seconds at which the phase is minimum.

Examples

`READ:PHVTIME:MEASview1:MINLOCATION?` might return `450.7E-9`, indicating the phase is minimum at 450.7 ns.

READ:PHVTime[:MEASview<y>]:OGRAm? (Query Only)

Returns the ogram trace data. The line is selected using the `TRACe:PHVTime:MEASview<y>:OGRAm:SELect:LINE` command.

Conditions

Applicable measurement view must be active.

Group

Read commands

Syntax

`READ:PHVTime[:MEASview<y>]:OGRAm?`

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional. By default y is 1.

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in <num_byte>.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the phase at the n^{th} data point in degrees (deg), represented as a 4-byte little-endian IEEE 488.2 floating-point value.

The unit can be changed by the `[SENSe]:POWer:UNITs` command.

Examples

`READ:PHVTime:MEASview1:OGRAm?` might return #43204xxxx... (3204-byte data) for the ogram trace.

READ:PHVTime[:MEASview<y>]:RESult? (Query Only)

Returns the Phase versus Time measurement results.

Conditions

Measurement views: Phase versus Time

Group

Read commands

Syntax

`READ:PHVTime[:MEASview<y>]:RESult?`

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

`<max>,<max_time>,<min>,<min_time>`

Where

`<max>::<NRf>` is the maximum phase in degrees.

`<max_time>::<NRf>` is the time in seconds at which the phase is maximum.

`<min>::<NRf>` is the minimum phase in degrees.

<min_time>::<NRf> is the time in seconds at which the phase is minimum.

Examples

READ:PHVTIME:MEASview1:RESULT? might return 153.8,175.3E-9,-176.3,450.7E-9, indicating the maximum phase is 153.8 ° at 175.3 ns and the minimum phase is -176.3 ° at 450.7 ns.

READ:PHVTime[:MEASview<y>]:TRACe:BITMap? (Query Only)

Acquires a waveform and then returns trace waveform data of the bitmap trace in the Phase versus Time measurement.

Conditions

Measurement views: Phase versus Time

Group

Read commands

Syntax

READ:PHVTime[:MEASview<y>]:TRACe:BITMap?

Arguments

None

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Returns

The format is a binary block of 32 bit floating point values. Each value ranges from 0 to 1.0, where 1.0 represents 100%.

Examples

READ:PHVTIME:TRACE:BITMAP? might return #42004xxxx... (2004-byte data) for the waveform data of the bitmap trace.

READ:PM:PNEGative? (Query Only)

Returns the negative peak phase deviation (–Pk) in the PM measurement.

Conditions

Measurement views: PM

Group

Read commands

Syntax

READ:PM:PNEGative?

Arguments

None

Returns

<-Pk>::<NRf> is the negative peak phase deviation in degrees.

Examples

`READ:PM:PNEGATIVE?` might return `-23.42`, indicating the positive peak phase deviation is –23.42 °.

READ:PM:PPOSitive? (Query Only)

Returns the positive peak phase deviation (+Pk) in the PM measurement.

Conditions

Measurement views: PM

Group

Read commands

Syntax

`READ:PM:PPOSitive?`

Arguments

None

Returns

`<+Pk>::<NRf>` is the positive peak phase deviation in degrees.

Examples

`READ:PM:PPOSITIVE?` might return `26.87`, indicating the positive peak phase deviation is 26.87°.

READ:PM:PTPeak? (Query Only)

Returns the peak-peak phase deviation (Pk-Pk) in the PM measurement.

Conditions

Measurement views: PM

Group

Read commands

Syntax

`READ:PM:PTPeak?`

Arguments

None

Returns

`<Pk-Pk>::<NRf>` is the peak-peak phase deviation in degrees.

Examples

`READ:PM:PTPEAK?` might return `46.34`, indicating the peak-peak phase deviation is 46.34 °.

READ:PM:RESult? (Query Only)

Returns the PM measurement results.

Conditions

Measurement views: PM

Group

Read commands

Syntax

`READ:PM:RESult?`

Arguments

None

Returns

`<+Pk>,<-Pk>,<RMS>,<Pk-Pk>`

Where

`<+Pk>::<NRf>` is the positive peak phase deviation in degrees.

`<-Pk>::<NRf>` is the negative peak phase deviation in degrees.

`<RMS>::<NRf>` is the RMS phase deviation in degrees.

`<Pk-Pk>::<NRf>` is the peak-peak phase deviation in degrees.

Examples

`READ:PM:RESULT?` might return `22.89,-23.45,15.12,46.34`.

READ:PM:RMS? (Query Only)

Returns the RMS phase deviation in the PM measurement.

Conditions

Measurement views: PM

Group

Read commands

Syntax

`READ:PM:RMS?`

Arguments

None

Returns

`<RMS>::<NRf>` is the RMS phase deviation in degrees.

Examples

`READ:PM:RMS?` might return `15.12`, indicating the RMS frequency deviation is 15.12 °.

READ:PNOise:ALL? (Query Only)

Returns all results of the phase noise measurement.

Conditions

Measurement views: Phase noise

Group

Read commands

Syntax

`READ:PNOise:ALL?`

Arguments

None

Returns

`<Cpower>,<Ferror>,<Pnoise>,<Tjitter>,<Rjitter>,<RFM>`

Where

`<Cpower>::<NRf>` is the carrier power in dBm.

`<Ferror>::<NRf>` is the frequency error in Hz.

`<Pnoise>::<NRf>` is the RMS phase noise in degrees.

`<Tjitter>::<NRf>` is the total jitter in seconds.

`<Rjitter>::<NRf>` is the random jitter in seconds.

`<RFM>::<NRf>` is the residual FMin Hz.

Examples

`READ:PNOISE:ALL?` might return `-9.455,1.235E+6,51.43,2.312E-9,4.178E-9,14.58`, indicating

Carrier power: -9.455 dBm,

Frequency error: 1.235 MHz,

RMS phase noise: 51.43 °,

Total jitter: 2.312 ns,

Random jitter: 4.178 ns, and

Residual FM: 14.58 Hz.

READ:PNOise:CARRier:FERRor? (Query Only)

Returns the carrier frequency error in the phase noise measurement.

Conditions

Measurement views: Phase noise

Group

Read commands

Syntax

READ:PNOise:CARRier:FERRor?

Arguments

None

Returns

<NRf> Carrier frequency error in Hz.

Examples

READ:PNOISE:CARRIER:FERROR? might return 1.235E+6, indicating that the carrier frequency error is 1.235 MHz.

READ:PNOise:CARRier:POWer? (Query Only)

Returns the carrier power in the phase noise measurement.

Conditions

Measurement views: Phase noise

Group

Read commands

Syntax

READ:PNOise:CARRier:POWer?

Arguments

None

Returns

<NRf> Carrier power in dBm.

The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples

READ:PNOISE:CARRIER:POWER? might return -9.455, indicating that the carrier power is -9.455 dBm.

READ:PNOise:JITTer? (Query Only)

Returns the jitter in the phase noise measurement.

Conditions

Measurement views: Phase noise

Group

Read commands

Syntax

READ:PNOise:JITTer?

Arguments

None

Returns

<NRf> Jitter in seconds.

Examples

READ:PNOISE:JITTER? might return 2.312E-9, indicating that the jitter is 2.312 ns.

READ:PNOise:RESidual:FM? (Query Only)

Returns the residual FM in the phase noise measurement.

Conditions

Measurement views: Phase noise

Group

Read commands

Syntax

READ:PNOise:RESidual:FM?

Arguments

None

Returns

<NRf> Residual FM in Hz.

Examples

READ:PNOISE:RESIDUAL:FM? might return 14.58, indicating that the residual FM is 14.58 Hz.

READ:PNOise:RMS:PNOise? (Query Only)

Returns the RMS phase noise in the phase noise measurement.

Conditions

Measurement views: Phase noise

Group

Read commands

Syntax

READ:PNOise:RMS:PNOise?

Arguments

None

Returns

<NRf> RMS phase noise in degrees.

Examples

READ:PNOISE:RMS:PNOISE? might return 51.43, indicating that the RMS phase noise is 51.43 °.

READ:PNOise:SPECtrum<x>:X? (Query Only)

Returns the frequencies of the specified trace.

The parameter <x> = 1 and 2, representing Trace 1 and Trace 2, respectively.

Conditions

Measurement views: Phase noise

Group

Read commands

Syntax

`READ:PNOise:SPECtrum<x>:X?`

Arguments

None

Returns

`#<num_digit><num_byte><x(1)><x(2)>...<x(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<x(n)>` is the frequency (Hz) at the nth point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`READ:PNOISE:SPECTRUM1:X?` might return `#516020xxxx...` (16020-byte data) for the frequencies of Trace 1.

READ:PNOise:SPECtrum<x>[:Y]? (Query Only)

Returns the phase noise values of the specified trace.

The parameter <x> = 1 and 2, representing Trace 1 and Trace 2, respectively.

Conditions

Measurement views: Phase noise

Group

Read commands

Syntax

`READ:PNOise:SPECtrum<x>[:Y]?`

Arguments

None

Returns

`#<num_digit><num_byte><y(1)><y(2)>...<y(n)>`

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<y(n)> is the phase noise (dBc/Hz) at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

READ:PNOISE:SPECTRUM1:Y might return #516020xxxx... (16020-byte data) for the phase noise values of Trace 1.

READ:PNOise:SPECtrum<x>:XY? (Query Only)

Returns the frequency and phase noise pairs of the specified trace.

The parameter <x> = 1 and 2, representing Trace 1 and Trace 2, respectively.

Conditions

Measurement views: Phase noise

Group

Read commands

Syntax

READ:PNOise:SPECtrum<x>:XY?

Arguments

None

Returns

#<num_digit><num_byte><x(1)><y(1)><x(2)><y(2)>...<x(n)><y(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<x(n)><y(n)> is the frequency (Hz) and phase noise (dBc/Hz) pair at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

READ:PNOISE:SPECTRUM1:XY? might return #516020xxxx... (16020-byte data) for the frequency and phase noise pairs of the Trace 1.

READ:PULSe:CUMulative:HISTogram[:MEASview<y>]:OUTLier:HIGHer? (Query Only)

Returns the outliers value exceeding the upper limit in pulse cumulative histogram.

Conditions

Measurement views: Cumulative Histogram with the specified Measurement View Number

Group

Read commands

Syntax

READ:PULSe:CUMulative:HISTogram[:MEASview<y>]:OUTLier:HIGHer?

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf> Number of outliers above higher limit

Examples

READ:PULSE:CUMULATIVE:HISTOGRAM:MEASview1:OUTLIER:HIGHER? might return 0.000, which is the value for the outliers that have exceeded the upper limit in the pulse cumulative histogram.

READ:PULSe:CUMulative:HISTogram[:MEASview<y>]:OUTLier:LOWer? (Query Only)

Returns the outliers value below the lower limit in pulse cumulative histogram.

Conditions

Measurement views: Cumulative Histogram with the specified Measurement View Number

Group

Read commands

Syntax

READ:PULSe:CUMulative:HISTogram[:MEASview<y>]:OUTLier:LOWer?

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf> Number of outliers below lower limit

Examples

READ:PULSE:CUMULATIVE:HISTOGRAM:MEASview1:OUTLIER:LOWER? might return 0.000, which is the value for the outliers below the lower limit in the pulse cumulative histogram.

READ:PULSe:CUMulative:HISTogram[:MEASview<y>]:PULse:COUNt? (Query Only)

Returns the current events and status conditions for the Cumulative Histogram display.

Conditions

Measurement views: Cumulative Histogram with the specified Measurement View Number

Group

Read commands

Syntax

READ:PULSe:CUMulative:HISTogram[:MEASview<y>]:EVENts?

Arguments

None

Returns

<ecode>,"<edesc>[<einfo]"{,<ecode>,"<edesc>[:<einfo>]"}

Where <ecode>::<NR1> is the error/event code (-32768 to 32767).

<edesc>::<string> is the description of the error/event.

<einfo>::<string> is the additional information on the error/event. If there is no error, then response is 0, "No events to report".

Examples

READ:PULSe:CUMulative:HISTogram:MEASview1:EVENTS might return 0, "No events to report".

READ:PULSe:CUMulative:HISTogram[:MEASview<y>]:PULse:TOTal? (Query Only)

Returns the total number of pulses in the pulse cumulative histogram.

Conditions

Measurement views: Cumulative Histogram with the specified Measurement View Number

Group

Read commands

Syntax

READ:PULSe:CUMulative:HISTogram[:MEASview<y>]:PULse:TOTal?

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf> Total number of pulses

Examples

READ:PULSE:CUMULATIVE:HISTOGRAM:MEASview1:PULSE:TOTAL? might return 16 for total number pulses in the pulse cumulative histogram.

READ:PULSe:CUMulative:STATistics[:MEASview<y>] (Query Only)

Returns the PulseCount, Max, Max TimeStamp, Min, Min TimeStamp, Peak to Peak, Average and Standard Deviation values for the selected measurement in pulse cumulative statistics.

Conditions

Measurement views: Pulse Cumulative Statistics

Group

Read commands

Syntax

READ:PULSe:CUMulative:STATistics[:MEASview<y>]? <arg1><arg2>

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

<arg1> = {AVERage| PPOWer| ATX| WIDTh| RISE| FALL| RINTerval| RRATe|DUTPct| DUTRatio| RIPPle| RIPDb| DROop| DRODb| OVERshoot| OVEDb|PPPHase| PPFRequency| RMSFreqerror| MFReqerror| RMSPherror| MPHerror|FRDeviation| FDELta| PHDeviation| IRAMplitude| IRTime| PPPD| PPFD| FABS}

<arg2> = {TPCOunt | MAX | MAXTs | MIN | MINTs | PK2PK | AVG | SD | ALL}

Returns

<puls_cnt> | <max> | <max_timstmp> | <min> | <min_timstmp> | <pk_pk> |<avg> | <std_dev> | <all>

For one of the following selected measurements:

<avg_on> | <pk_pwr> | <avg_tx> | <width> | <rise> | <fall> | <rep_int> |<rep_rate> | <duty_factor_pcnt> | <duty_fact_ratio> | <ripple> | <ripple_db> |<droop> | <droop_db> | <ovrshoot> | <ovrshoot_db> | <pp_phase> | <pp_freq> |<rms_phase_err> | <rms_freq_err> | <p2p_phase> | <p2p_freq> | <pk_phase_err>| <pk_freq_err> | <abs_freq> | <freq_dev> | <phase_dev> | <dlta_freq> |<imp_resp_amp> | <imp_resp_time>

where

<avg_on>::<NRf> is the average power transmitted during pulse on

<pk_pwr>::<NRf> is the Maximum power during pulse on

<avg_tx>:: <NRf> is the average power transmitted

<width>::<NRf> is the time from the rising edge to the falling edge at the −3 dB / −6 dB level (50%) of the user selected 100% level

<rise>::<NRf> is the time required for a signal to rise from 10% to 90% (or 20% to 80%) of the user selected 100% level

<fall> ::<NRf> is the time required for a signal to fall from 90% to 10% (or 80% to 20%) of the user selected 100% level

<rep_int>::<NRf> is the time from a pulse rising edge to the next pulse rising edge

<rep_rate> ::<NRf> is the inverse of repetition interval

<duty_factor_pcnt>::<NRf> is the ratio of the width to the pulse period, expressed as a percentage

<duty_fact_ratio>::<NRf> is the ratio of the pulse width to the pulse period

<ripple>::<NRf> is the Ripple is the peak-to-peak ripple on the pulse top

<ripple_db>::<NRf> is the Ripple measurement expressed in dB

<droop>::<NRf> is the Droop is the power difference between the beginning and the end of the pulse On time

<droop_db>::<NRf> is the Droop measurement expressed in dB

<ovrshoot>::<NRf> is the amount by which the signal exceeds the 100% level on the pulse rising edge

<ovrshoot_db>::<NRf> is the Overshoot measurement expressed in dB

<pp_phase>::<NRf> is the phase difference between the selected pulse and the previous pulse in the analysis window

<pp_freq> ::<NRf> is the difference between the frequency of the current pulse and frequency of the previous pulse

<rms_phase_err>::<NRf> is the RMS average of the Phase vs Time trace, computed over the Measurement Time

<rms_freq_err>::<NRf> is the RMS average of the Freq Error vs. Time trace

<p2p_phase>::<NRf> is the frequency difference between the current pulse and the first pulse

<p2p_freq>::<NRf> is the frequency difference between the current pulse and immediate previous pulse

<pk_phase_err>::<NRf> is the maximum phase error

<abs_freq>::<NRf> is the absolute pulse frequency measured at a specific point

<freq_dev>::<NRf> is the difference between the maximum and minimum measured values of the signal frequency during the Measurement Time

<phase_dev>::<NRf> is the Phase Deviation difference between the maximum and minimum Phase values measured during the On time of a pulse

<dlta_freq>::<NRf> is the difference from the measurement frequency to each pulse frequency

<imp_resp_amp>::<NRf> is the difference in dB between the levels of the main lobe and highest side lobe

<imp_resp_time>::<NRf> is the difference in time between the main lobe and highest side lobe

Examples

READ:PULSe:CUMulative:STATistics:MEASview1? AVERage, MAX might return –1.3594852284, indicating the max value for Average ON is –1.3594852284.

READ:PULSe:OGRAm[:MEASview<y>]:MAX:FRAMe:COUNt (Query Only)

Returns the frame count value in the Pulse-Ogram display.

Conditions

Measurement views: Pulse-Ogram

Group

Read commands

Syntax

READ:PULSe:OGRAm[:MEASview<y>]:MAX:FRAMe:COUNt?

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Returns

<NRf>

Examples

READ:PULSe:OGRAm:MEASview1:MAX:FRAMe:COUNt? might return 20, indicating the total number of fast frames in the acquisition of Pulse-Ogram is 20.

READ:PULSe:OGRAm[:MEASview<y>]:RBW (Query Only)

Returns the resolution bandwidth value only when Frequency vs Time display is opened in the Pulse-Ogram display.

Conditions

Measurement views: Pulse-Ogram

Group

Read commands

Syntax

`READ:PULSe:OGRAm[:MEASview<y>]:RBW?`

Arguments

`<y>` = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Returns

`<NRf>`

Examples

`READ:PULSe:OGRAm:MEASview1:RBW?` might return `222.9199886322E+3`, indicating the RBW value is 222.9199886322E+3 in Pulse-Ogram.

READ:PULSe[:RESult][:MEASview<y>]:ADVanced (Query Only)

Returns the Pulse Count, Max, Max at Pulse number, Min, Min at Pulse number, Peak to Peak, Average and Standard Deviation values for the chosen measurement in pulse table.

Conditions

Measurement views: All

Group

Read commands

Syntax

`READ:PULSe[:RESult][:MEASview<y>]:ADVanced <arg1>,<arg2>`

Arguments

`arg1` can be one of the following measurements:

`{AVERage| PPOWer| ATX|WIDTh| RISE| FALL| RINTerval| RRATe|DUTPct| DUTRatio|RIPPle| RIPDb| DROop| DRODb| OVERshoot|OVEDb| PPPHase| PPFRequency| RMSFreqerror| MFReqerror|RMSPherror| MPHerror| FRDeviation| FDELta| PHDeviation|IRAMplitude|IRTime|PPPD|PPFD|FABS}`

`arg2` can be one of the following statistics results:

`{TPCOunt|MAX|MAXTs|MIN|MINTs|PK2PK|AVG|SD|ALL}`

Returns

`<puls_cnt> | <max> | <min> | <pk_pk> | <avg> | <std_dev> | <all>`

For one of the following selected measurements, where

<avg_on> | <pk_pwr> | <avg_tx> | <width> | <rise> | <fall> | <rep_int> | <rep_rate> | <duty_factor_pcnt> | <duty_fact_ratio> | <ripple> | <ripple_db> | <droop> | <droop_db> | <ovrshoot> | <ovrshoot_db> | <pp_phase> | <pp_freq> | <rms_phase_err> | <rms_freq_err> | <p2p_phase> | <p2p_freq> | <pk_phase_err> | <pk_freq_err> | <abs_freq> | <freq_dev> | <phase_dev> | <dlta_freq> | <imp_resp_amp> | <imp_resp_time>|<time> for the given pulse number

where

<avg_on>::<NRf> is the average power transmitted during pulse on.

<pk_pwr>::<NRf> is the Maximum power during pulse on

<avg_tx>:: <NRf> is the average power transmitted

<width>::<NRf> is the time from the rising edge to the falling edge at the −3 dB / −6 dB level (50%) of the user selected 100% level

<rise>::<NRf> is the time required for a signal to rise from 10% to 90% (or 20% to 80%) of the user selected 100% level

<fall> ::<NRf> is the time required for a signal to fall from 90% to 10% (or 80% to 20%) of the user selected 100% level

<rep_int>::<NRf> is the time from a pulse rising edge to the next pulse rising edge

<rep_rate> ::<NRf> is the inverse of repetition interval

<duty_factor_pcnt>::<NRf> is the ratio of the width to the pulse period, expressed as a percentage

<duty_fact_ratio>::<NRf> is the ratio of the pulse width to the pulse period

<ripple>::<NRf> is the Ripple is the peak-to-peak ripple on the pulse top

<ripple_db>::<NRf> is the Ripple measurement expressed in dB

<droop>::<NRf> is the Droop is the power difference between the beginning and the end of the pulse On time

<droop_db>::<NRf> is the Droop measurement expressed in dB

<ovrshoot>::<NRf> is the amount by which the signal exceeds the 100% level on the pulse rising edge

<ovrshoot_db>::<NRf> is the Overshoot measurement expressed in dB

<pp_phase>::<NRf> is the phase difference between the selected pulse and the previous pulse in the analysis window

<pp_freq> ::<NRf> is the difference between the frequency of the current pulse and frequency of the previous pulse

<rms_phase_err>::<NRf> is the RMS average of the Phase vs Time trace, computed over the Measurement Time

<rms_freq_err>::<NRf> is the RMS average of the Freq Error vs.Time trace

<p2p_phase>::<NRf> is the frequency difference between the current pulse and the first pulse

<p2p_freq>::<NRf> is the frequency difference between the current pulse and immediate previous pulse

<pk_phase_err>::<NRf> is the maximum phase error

<abs_freq>::<NRf> is the absolute pulse frequency measured at a specific point

<freq_dev>::<NRf> is the difference between the maximum and minimum measured values of the signal frequency during the Measurement Time

<phase_dev>::<NRf> is the Phase Deviation difference between the maximum and minimum Phase values measured during the On time of a pulse

<dlta_freq>::<NRf> is the difference from the measurement frequency to each pulse frequency

<imp_resp_amp>::<NRf> is the difference in dB between the levels of the main lobe and highest side lobe

<imp_resp_time>::<NRf> is the difference in time between the main lobe and highest side lobe

<time>::<NRf> is the dtime in seconds relative to the time reference point in the first acquisition record in the data set.

Examples

`READ:PULSe:RESult:MEASview2:ADVanced AVERage, MAX` might return `-1.3594852284` for Average ON value in pulse cumulative statistics

READ:PULSe[:RESult][:MEASview<y>]:ADVanced:RANGe (Query only)

Queries the Pulse Count, Max, Max TimeStamp, Min, Min TimeStamp, Peak to Peak, Average, and Standard Deviation values for the select measurement in the Pulse Table display.

Conditions

Measurement views: Pulse Table

Group

Fetch commands

Syntax

`READ:PULSe[:RESult][:MEASview<y>]:ADVanced:RANGe <arg1><arg2><arg3>`

Arguments

`<arg1>={Start_Pulse_number}`

`<arg2>={End_Pulse_number}`

`<arg3>= {AVERage | PPOWer | ATX | WIDTh | RISE | FALL | RINTerval | RRATe | DUTPct | DUTRatio |RIPPle | RIPDb | DROop | DRODb | OVERshoot | OVEDb | PPPHase | PPFRequency | RMSFreqerror | MFReqerror | RMSPherror |MPHerror | FRDeviation | FDELta | PHDeviation | IRAMplitude|TIME | IRTime | PPPD| PPFD | FABS}`

Returns

`<puls_cnt> | <max> | <max_timstmp> | <min> | <min_timstmp> | <pk_pk> | <avg> | <std_dev> | <all>`

For one of the following selected measurements:

`<avg_on> | <pk_pwr> | <avg_tx> | <width> | <rise> | <fall> | <rep_int> | <rep_rate> | <duty_factor_pcnt> | <duty_fact_ratio> | <ripple> | <ripple_db> | <droop> | <droop_db> | <ovrshoot> | <ovrshoot_db> | <pp_phase> | <pp_freq> | <rms_phase_err> | <rms_freq_err> | <p2p_phase> | <p2p_freq> | <pk_phase_err> | <pk_freq_err> | <abs_freq> | <freq_dev> | <phase_dev> | <dlta_freq> | <imp_resp_amp> | <imp_resp_time>|<time>` for the given pulse number.

where

`<avg_on>::<NRf>` is the average power transmitted during pulse on.

`<pk_pwr>::<NRf>` is the Maximum power during pulse on.

`<avg_tx>:: <NRf>` is the average power transmitted.

`<width>::<NRf>` is the time from the rising edge to the falling edge at the −3 dB or −6 dB level (50%) of the user selected 100% level.

`<rise>::<NRf>` is the time required for a signal to rise from 10% to 90% or (20% to 80%) of the user selected 100% level.

`<fall> ::<NRf>` is the time required for a signal to fall from 90% to 10% or (80% to 20%) of the user selected 100% level.

`<rep_int>::<NRf>` is the time from a pulse rising edge to the next pulse rising edge.

<rep_rate> ::<NRf> is the inverse of repetition interval.

<duty_factor_pcnt>::<NRf> is the ratio of the width to the pulse period, expressed as a percentage.

<duty_fact_ratio>::<NRf> is the ratio of the pulse width to the pulse period.

<ripple>::<NRf> is the Ripple is the peak-to-peak ripple on the pulse top.

<ripple_db>::<NRf> is the Ripple measurement expressed in dB.

<droop>::<NRf> is the Droop is the power difference between the beginning and the end of the pulse On time.

<droop_db>::<NRf> is the Droop measurement expressed in dB.

<ovrshoot>::<NRf> is the amount by which the signal exceeds the 100% level on the pulse rising edge.

<ovrshoot_db>::<NRf> is the Overshoot measurement expressed in dB.

<pp_phase>::<NRf> is the phase difference between the selected pulse and the previous pulse in the analysis window.

<pp_freq> ::<NRf> is the difference between the frequency of the current pulse and frequency of the previous pulse.

<rms_phase_err>::<NRf> is the RMS average of the Phase versus Time trace, computed over the Measurement Time.

<rms_freq_err>::<NRf> is the RMS average of the Freq error versus Time trace.

<p2p_phase>::<NRf> is the frequency difference between the current pulse and the first pulse.

<p2p_freq>::<NRf> is the frequency difference between the current pulse and immediate previous pulse.

<pk_phase_err>::<NRf> is the maximum phase error.

<abs_freq>::<NRf> is the absolute pulse frequency measured at a specific point.

<freq_dev>::<NRf> is the difference between the maximum and minimum measured values of the signal frequency during the Measurement Time.

<phase_dev>::<NRf> is the Phase deviation difference between the maximum and minimum Phase values measured during the On time of a pulse.

<dlta_freq>::<NRf> is the difference from the measurement frequency to each pulse frequency.

<imp_resp_amp>::<NRf> is the difference in dB between the levels of the main lobe and highest side lobe.

<imp_resp_time>::<NRf> is the difference in time between the main lobe and highest side lobe.

<time>::<NRf> is the dtime in seconds relative to the time reference point in the first acquisition record in the data set.

Example

READ:PULSE[:RESULT]:MEASview1:ADVANCED:RANGe 2048,2060,AVERage might return -3,23.8467659801E-3,23.8566566259E-3,23.8667335361E-3,23.8771531731E-3,23.8866508007E-3,23.89664 indicating the average ON value for the pulse number from 2048 to 2060.

READ:PULSe:STATistics[:MEASview<y>]:ADVanced (Query Only)

Returns the Pulse Count, Max, Max at Pulse number, Min, Min at Pulse number, Peak to Peak, Average and Standard Deviation values for the chosen measurement in Pulse Statistics display.

Conditions

Measurement views: Pulse Statistics

Group

Read commands

Syntax

```
READ:PULSe:STATistics:[:MEASview<y>]:ADVanced <arg1>,<arg2>
READ:PULSe:STATistics:[:MEASview<y>]:AD;Vanced?
```

Arguments

<arg1> = {AVERage | PPOWer | ATX | WIDTh | RISE | FALL | RINTerval | RRATe | DUTPct | DUTRatio |RIPPle | RIPDb | DROop | DRODb | OVERshoot | OVEDb | PPPHase | PPFRequency | RMSFreqerror | MFReqerror | RMSPherror |MPHerror | FRDeviation | FDELta | PHDeviation | IRAMplitude|TIME | IRTime | PPPD| PPFD | FABS}

<arg2> = {TPCOunt |MAX |MIN |PK2PK |AVG |SD |ALL}

Returns

<puls_cnt> | <max> | <min> | <pk_pk> | <avg> | <std_dev> | <all>

For one of the following selected measurements:

<avg_on> | <pk_pwr> | <avg_tx> | <width> | <rise> | <fall> | <rep_int> | <rep_rate> | <duty_factor_pcnt> | <duty_fact_ratio> | <ripple> | <ripple_db> | <droop> | <droop_db> | <ovrshoot> | <ovrshoot_db> | <pp_phase> | <pp_freq> | <rms_phase_err> | <rms_freq_err> | <p2p_phase> | <p2p_freq> | <pk_phase_err> | <pk_freq_err> | <abs_freq> | <freq_dev> | <phase_dev> | <dlta_freq> | <imp_resp_amp> | <imp_resp_time>|<time> for the given pulse number

where

<avg_on>::<NRf> is the average power transmitted during pulse on.

<pk_pwr>::<NRf> is the Maximum power during pulse on

<avg_tx>:: <NRf> is the average power transmitted

<width>::<NRf> is the time from the rising edge to the falling edge at the -3 dB /−6 dB level (50%) of the user selected 100% level

<rise>::<NRf> is the time required for a signal to rise from 10% to 90% (or 20% to 80%) of the user selected 100% level

<fall> ::<NRf> is the time required for a signal to fall from 90% to 10% (or 80% to 20%) of the user selected 100% level

<rep_int>::<NRf> is the time from a pulse rising edge to the next pulse rising edge

<rep_rate> ::<NRf> is the inverse of repetition interval

<duty_factor_pcnt>::<NRf> is the ratio of the width to the pulse period, expressed as a percentage

<duty_fact_ratio>:: <NRf> is the ratio of the pulse width to the pulse period

<ripple>::<NRf> is the Ripple is the peak-to-peak ripple on the pulse top

<ripple_db>::<NRf> is the Ripple measurement expressed in dB

<droop>::<NRf> is the Droop is the power difference between the beginning and the end of the pulse On time

<droop_db>::<NRf> is the Droop measurement expressed in dB

<ovrshoot>::<NRf> is the amount by which the signal exceeds the 100% level on the pulse rising edge

<ovrshoot_db>::<NRf> is the Overshoot measurement expressed in dB

<pp_phase>::<NRf> is the phase difference between the selected pulse and the previous pulse in the analysis window

<pp_freq> ::<NRf> is the difference between the frequency of the current pulse and frequency of the previous pulse

<rms_phase_err>::<NRf> is the RMS average of the Phase vs Time trace, computed over the Measurement Time

<rms_freq_err>::<NRf> is the RMS average of the Freq Error vs.Time trace

<p2p_phase>::<NRf> is the frequency difference between the current pulse and the first pulse

<p2p_freq>::<NRf> is the frequency difference between the current pulse and immediate previous pulse

<pk_phase_err>::<NRf> is the maximum phase error

<abs_freq>::<NRf> is the absolute pulse frequency measured at a specific point

<freq_dev>::<NRf> is the difference between the maximum and minimum measured values of the signal frequency during the Measurement Time

<phase_dev>::<NRf> is the Phase Deviation difference between the maximum and minimum Phase values measured during the On time of a pulse

<dlta_freq>::<NRf> is the difference from the measurement frequency to each pulse frequency

<imp_resp_amp>::<NRf> is the difference in dB between the levels of the main lobe and highest side lobe

<imp_resp_time>::<NRf> is the difference in time between the main lobe and highest side lobe

<time>::<NRf> is the dtime in seconds relative to the time reference point in the first acquisition record in the data set

Examples

READ:PULSE:STATISTICS:MEASview1:ADVANCED RISE, MAX might return 7.9786177665E-9,"-", indicating the maximum rise value is 7.9786177665E-9.

READ:PULSe[:RESult][:MEASview<y>]:ATX (Query Only)

Returns the average transmitted power in the pulse table.

Conditions

Measurement views: Pulse table

Group

Read commands

Syntax

READ:PULSe[:RESult][:MEASview<y>]:ATX?

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<first_pulse_num>,<ATX(1)>,< ATX(2)>,...<ATX(n)>

Where

<first_pulse_num>::<NR1> is the first pulse number.

<ATX(n)>::<NRf> is the average transmitted power for the pulse with the number of [first_pulse_num + n - 1] in dBm.

The unit can be changed to watts by the [SENSe]:POWer:UNITs command.

This measurement gives valid results only for power units (Watts, dBm, dBuW, dBW).

Examples

READ:PULSe:RESult:MEASview2:ATX? might return 0,-18.57,-18.73,-18.20, -18.53 for Pulse 0 to 3.

READ:PULSe[:RESult][:MEASview<y>]:AVERage (Query Only)

Returns the average on power in the pulse table.

Conditions

Measurement views: Pulse table

Group

Read commands

Syntax

`READ:PULSe[:RESult][:MEASview<y>]:AVERage?`

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<first_pulse_num>,<avg(1)>,< avg(2)>,...<avg(n)>`

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

`<avg(n)>::<NRf>` is the average on power for the pulse with the number of [first_pulse_num + n - 1] in dBm.

The unit can be changed by the [SENSe]:POWer:UNITs (on page 2269) command.

This measurement gives valid results only for power units (Watts, dBm, dBuW, dBW).

Examples

`READ:PULSe:RESult:MEASview2:AVERage?` might return `0,-2.354,-2.368,-2.343,-2.358` for Pulse 0 to 3.

READ:PULSe[:RESult][:MEASview<y>]:DRODb (Query Only)

Returns the droop in dB in the pulse table.

Conditions

Measurement views: Pulse table

Group

Read commands

Syntax

`READ:PULSe[:RESult][:MEASview<y>]:DRODb?`

Related Commands

READ:PULSe[:RESult][:MEASview<y>]:DROop? (on page 1775)

Arguments

None

Returns

<first_pulse_num>,<droop(1)>,<droop(2)>,...<droop(n)>

Where

<first_pulse_num>::<NR1> is the first pulse number.

<droop(n)>::<NRf> is the droopdB for the pulse number [first_pulse_num + n - 1] in percent (%) of watts or volts.

Examples

READ:PULSe:RESult:MEASview2:DRODb? might return 1,-0.4,0.32,0.2,-0.1 for Pulse 1 to 4.

READ:PULSe[:RESult][:MEASview<y>]:DROop (Query Only)

Returns the average on power in the pulse table.

Conditions

Measurement views: Pulse table

Group

Read commands

Syntax

READ:PULSe[:RESult][:MEASview<y>]:DROop?

Related Commands

READ:PULSe[:RESult][:MEASview<y>]:DRODb? (on page 1774)

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<first_pulse_num>,<droop(1)>,<droop(2)>,...<droop(n)>

Where

<first_pulse_num>::<NR1> is the first pulse number.

<droop(n)>::<NRf> is the droop for the pulse number [first_pulse_num + n - 1] in percent (%) of watts or volts.

Examples

READ:PULSe:RESult:MEASview2:DROop? might return 0,-270.9E-3,-193.0E-3, -242.7E-3,-177.5E-3 for Pulse 0 to 3.

READ:PULSe[:RESult][:MEASview<y>]:DUTPct (Query Only)

Returns the duty factor (%) in the pulse table.

Conditions

Measurement views: Pulse table

Group

Read commands

Syntax

```
READ:PULSe[:RESult][:MEASview<y>]:DUTPct?
```

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<first_pulse_num>,<duty_pct(1)>,<duty_pct(2)>,...<duty_pct(n)>`

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

`<duty_pct(n)>::<NRf>` is the duty factor for the pulse with the number of [first_pulse_num + n - 1] in percent (%).

Examples

`READ:PULSe:RESult:MEASview2:DUTPct?` might return `0,28.94,28.96,29.00,29.01` for Pulse 0 to 3.

READ:PULSe[:RESult][:MEASview<y>]:DUTRatio (Query Only)

Returns the duty factor (ratio) in the pulse table.

Conditions

Measurement views: Pulse table

Group

Read commands

Syntax

```
READ:PULSe[:RESult][:MEASview<y>]:DUTRatio?
```

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<first_pulse_num>,<duty_ratio(1)>,<duty_ratio(2)>,… <duty_ratio(n)>`

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

`<duty_ratio(n)>::<NRf>` is the duty factor for the pulse with the number of [first_pulse_num + n - 1] (no unit).

Examples

`READ:PULSe:RESult:MEASview2:DUTRatio?` might return `0,289.4E-3,289.6E-3, 290.0E-3,290.1E-3` for Pulse 0 to 3.

READ:PULSe[:RESult][:MEASview<y>]:FABS (Query Only)

Returns the computed absolute frequency from the carrier in the pulse table.

Conditions

Measurement views: Pulse table

Group

Read commands

Syntax

`READ:PULSe[:RESult][:MEASview<y>]:FABS?`

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<first_pulse_num>,<abs_freq(1)>,<abs_freq(2),.....<abs_freq(n)>`

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

`<abs_freq(n)>::<NR1>` is the absolute frequency for the pulse with the number of [first_pulse_num + n — 1] in Hz.

Examples

`READ:PULSe:RESult:MEASview2:FABS?` might return `1.0000000000, 919.8726400000E+6, 1.1664642560E+9, 1.1866586880E+9, 1.1867939840E+9, 1.1882243840E+9, 933.2595840000E+6, 1.1729175040E+9, 1.1818649600E+9, 1.1868037120E+9, 1.1832893440E+9, 933.4762880000E+6, 1.1805880320E+9, 1.1844794880E+9, 1.1914785280E+9, 1.1929450240E` for is the number of the first pulse and the absolute frequency for each pulse.

READ:PULSe[:RESult][:MEASview<y>]:FALL (Query Only)

Returns the fall time in the pulse table.

Conditions

Measurement views: Pulse table

Group

Read commands

Syntax

`READ:PULSe[:RESult][:MEASview<y>]:FALL?`

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<first_pulse_num>,<fall(1)>,<fall(2)>,...<fall(n)>`

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

`<fall(n)>::<NRf>` is the fall time for the pulse with the number of [first_pulse_num + n - 1] in seconds.

Examples

READ:PULSe:RESult:MEASview2:FALL? might return 0,110.3E-9,90.45E-9, 95.03E-9,111.9E-9 for Pulse 0 to 3.

READ:PULSe[:RESult][:MEASview<y>]:FDELta (Query Only)

Returns the computed delta frequency from the carrier in the pulse table.

Conditions

Measurement views: Pulse table

Group

Read commands

Syntax

READ:PULSe[:RESult][:MEASview<y>]:FDELta?

Arguments

None

Returns

<first_pulse_num>,<freq_delta(1)>,<freq_delta(2)>,…<freq_delta(n)>

Where

<first_pulse_num>::<NR1> is the first pulse number.

<freq_delta(n)>::<NRf> is the frequency delta for the pulse with the number of [first_pulse_num + n - 1] in Hz.

Examples

READ:PULSe:RESult:MEASview2:FDELta? might return
1.0000000000,163.6692962646,82.8298492432,21.8995475769,
89.2963562012,48.1314239502,144.1430664063,52.5188140869,
19.9404506683,287.8957519531,110.4534988403,-110.1940994263,
73.9322433472,115.7962646484,164.6119384766,-42.2758331299, 223.0057525635,28.680925 for the first pulse and each pulse found.

READ:PULSe[:RESult][:MEASview<y>]:FRDeviation (Query Only)

Returns the frequency deviation in the pulse table.

Conditions

Measurement views: Pulse table

Group

Read commands

Syntax

READ:PULSe[:RESult][:MEASview<y>]:FRDeviation?

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<first_pulse_num>,<freq_dev(1)>,<freq_dev(2)>,...<freq_dev(n)>

Where

<first_pulse_num>::<NR1> is the first pulse number.

<freq_dev(n)>::<NRf> is the frequency deviation for the pulse with the number of [first_pulse_num + n - 1] in Hz.

Examples

READ:PULSe:RESult:MEASview2:FRDeviation? might return 1,740.6E+3,736.5E+3,718.3E+3,672.2E+3 for Pulse 1 to 4.

READ:PULSe[:RESult][:MEASview<y>]:IRAMplitude (Query Only)

Returns the pulse impulse response amplitude in the pulse table.

Conditions

Measurement views: Pulse table

Group

Read commands

Syntax

READ:PULSe[:RESult][:MEASview<y>]:IRAMplitude?

Returns

<first_pulse_num>,<amplitude(1)>,<amplitude(2)>,...<amplitude(n)>

Where

<first_pulse_num>::<NR1> is the first pulse number.

<amplitude(n)>::<NRf> is the sidelobe level, in dB, from the mainlobe for the pulse with the number of [first_pulse_num + n - 1].

Examples

READ:PULSe:RESult:MEASview2:IRAMplitude? might return 1.0000000000,-45.3,-47.5,-50.2 for Pulse 1 through 3.

READ:PULSe[:RESult][:MEASview<y>]:IRTime (Query Only)

Returns the impulse response time in the pulse table.

Conditions

Measurement views: Pulse table

Group

Read commands

Syntax

READ:PULSe[:RESult][:MEASview<y>]:IRTime?

Returns

<first_pulse_num>,<time(1)>,<time(2)>,...<time(n)>

Where

<first_pulse_num>::<NR1> is the first pulse number.

<time(n)>::<NRf> is the sidelobe offset time from the mainlobe for the pulse with the number of [first_pulse_num + n - 1] in seconds.

Examples

READ:PULSe:RESult:MEASview2:IRTime? might return 1.0000000000,806.6708687693E-9,-846.6595318168E-9,-873.3477443457E-9 for Pulse 1 to 3.

READ:PULSe[:RESult][:MEASview<y>]:MFReqerror (Query Only)

Returns the maximum frequency error in the pulse table.

Conditions

Measurement views: Pulse table

Group

Read commands

Syntax

READ:PULSe[:RESult][:MEASview<y>]:MFReqerror?

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<first_pulse_num>,<max_freq_err(1)>,<max_freq_err(2)>,...<max_freq_err(n)>

Where

<first_pulse_num>::<NR1> is the first pulse number.

<max_freq_err(n)>::<NRf> is the maximum frequency error for the pulse with the number of [first_pulse_num + n - 1] in Hz.

Examples

READ:PULSe:RESult:MEASview2:MFReqerror? might return 1,597.5E+3,675.8E+3,642.8E+3,598.2E+3 for Pulse 1 to 4.

READ:PULSe[:RESult][:MEASview<y>]:MPHerror (Query Only)

Returns the maximum phase error in the pulse table.

Conditions

Measurement views: Pulse table

Group

Read commands

Syntax

READ:PULSe[:RESult][:MEASview<y>]:MPHerror?

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<first_pulse_num>,<max_phase_err(1)>,<max_phase_err(2)>,...<max_phase_err(n)>

Where

<first_pulse_num>::<NR1> is the first pulse number.

<max_phase_err(n)>::<NRf> is the maximum phase error for the pulse with the number of [first_pulse_num + n - 1] in degrees.

Examples

READ:PULSe:RESult:MEASview2:MPHerror? might return 1,-9.221,-8.413, -11.853,-10.258 for Pulse 1 to 4.

READ:PULSe[:RESult][:MEASview<y>]:OVERDb (Query Only)

Returns the overshoot in dB in the pulse table.

Conditions

Measurement views: Pulse table

Group

Read commands

Syntax

READ:PULSe[:RESult][:MEASview<y>]:OVERDb?

Related Commands

READ:PULSe[:RESult][:MEASview<y>]:OVERshoot? (on page 1782)

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<first_pulse_num>,<overshoot(1)>,<overshoot(2)>,...<overshoot(n)>

Where

<first_pulse_num>::<NR1> is the first pulse number.

<overshoot(n)>::<NRf> is the overshoot for the pulse with the number of [first_pulse_num + n - 1] in dB.

Examples

READ:PULSe:RESult:MEASview2:OVERDb? might return 1,1.2,2.3,1.0,0.2 for Pulse 1 to 4.

READ:PULSe[:RESult][:MEASview<y>]:OVERshoot (Query Only)

Returns the overshoot in the pulse table.

Conditions

Measurement views: Pulse table

Group

Read commands

Syntax

`READ:PULSe[:RESult][:MEASview<y>]:OVERshoot?`

Related Commands

READ:PULSe[:RESult][:MEASview<y>]:OVERDb? (on page 1781)

Arguments

None

Returns

`<first_pulse_num>,<overshoot(1)>,<overshoot(2)>,...<overshoot(n)>`

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

`<overshoot(n)>::<NRf>` is the overshoot for the pulse with the number of [first_pulse_num + n - 1] in percent (%) of watts or volts.

Examples

`READ:PULSe:RESult:MEASview2:OVERshoot?` might return `1,1.2,2.3,1.0,0.2` for Pulse 1 through 4.

READ:PULSe[:RESult][:MEASview<y>]:PHDeviation (Query Only)

Returns the phase deviation in the pulse table.

Conditions

Measurement views: Pulse table

Group

Read commands

Syntax

`READ:PULSe[:RESult][:MEASview<y>]:PHDeviation?`

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<first_pulse_num>,<phase_dev(1)>,<phase_dev(2)>,...<phase_dev(n)>`

Where

<first_pulse_num>::<NR1> is the first pulse number.

<phase_dev(n)>::<NRf> is the phase deviation for the pulse with the number of [first_pulse_num + n - 1] in degrees.

Examples

READ:PULSe:RESult:MEASview2:PHDeviation? might return 1,11.658,9.640,10.509,8.272 for Pulse 1 to 4.

READ:PULSe[:RESult][:MEASview<y>]:PPFD (Query Only)

Returns the computed pulse to pulse frequency difference from the carrier in the pulse table.

Conditions

Measurement views: Pulse table

Group

Read commands

Syntax

READ:PULSe[:RESult][:MEASview<y>]:PPFD

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<first_pulse_num>,<ppfd(1)>,<ppfd(2),.....<ppfd(n)>

Where

<first_pulse_num>::<NR1> is the first pulse number.

<ppfd(n)>::<NRf> is the pulse to pulse frequency difference with the number of [first_pulse_num + n – 1] in Hz.

Examples

READ:PULSe:RESult:MEASview2:PPFD? might return 1.0000000000,"-", 246.5916160000E+6, 20.1943520000E+6, 135.2960000000E+3, 1.4304160000E+6, -254.9647520000E+6, 239.6579520000E+6, 8.9474560000E+6, 4.9386560000E+6, -3.5143840000E+6, -249.8130240000E+6, 247.1117760000E+6, 3.8915200000E+6, 6.9989120000E+6, 1.4665920000E+6 is the number of the first pulse and the pulse frequency difference for each pulse.

READ:PULSe[:RESult][:MEASview<y>]:PPFRequency (Query Only)

Returns the pulse-pulse carrier frequency in the pulse table.

Conditions

Measurement views: Pulse table

Group

Read commands

Syntax

READ:PULSe[:RESult][:MEASview<y>]:PPFRequency?

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<first_pulse_num>,<pp_freq(1)>,<pp_freq(2)>,...<pp_freq(n)>`

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

`<pp_freq(n)>::<NRf>` is the pulse-pulse carrier frequency for the pulse with the number of [first_pulse_num + n - 1] in Hz.

Examples

`READ:PULSe:RESult:MEASview2:PPFRequency?` might return `0,0.000,1.258E+3,-3.121E+3,1.862E+3` for Pulse 0 to 3.

READ:PULSe[:RESult][:MEASview<y>]:PPOWer (Query Only)

Returns the peak power in the pulse table.

Conditions

Measurement views: Pulse table

Group

Read commands

Syntax

`READ:PULSe[:RESult][:MEASview<y>]:PPOWer?`

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<first_pulse_num>,<pk_power(1)>,<pk_power(2)>,...<pk_power(n)>`

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

`<pk_power(n)>::<NRf>` is the peak power for the pulse with the number of [first_pulse_num + n - 1] in dBm.

The unit can be changed by the [SENSe]:POWer:UNITs command.

This measurement gives valid results only for power units(Watts, dBm, dBuW, dBW).

Examples

`READ:PULSe:RESult:MEASview2:PPOWer?` might return `0,-2.26,-2.27,-2.23,-2.25` for Pulse 0 to 3.

READ:PULSe[:RESult][:MEASview<y>]:PPPD (Query Only)

Returns the computed pulse to pulse phase difference from the carrier in the pulse table.

Conditions

Measurement views: Pulse table

Group

Read commands

Syntax

`READ:PULSe[:RESult][:MEASview<y>]:PPPD`

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<first_pulse_num>,<pppd(1)>,<pppd(2),.....<pppd(n)>`

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

`<pppd(n)>::<NRf>` is the pulse to pulse phase difference with the number of [first_pulse_num + n − 1] in Hz.

Examples

`READ:PULSe:RESult:MEASview2:PPPD?` might return `1.0000000000,"-", 96.2469482422, 24.9118175507, 5.8206896782, 16.7037734985, -151.8063964844, 100.7584228516,36.6761512756, -3.8614616394, 12.9028282166, -150.9820556641, 130.5854492188, 9.1259155273, -7.4177660942, 6.0613451004, -142.4224853516` is the number of the first pulse and the pulse to pulse phase difference for each pulse.

READ:PULSe[:RESult][:MEASview<y>]:PPPHase (Query Only)

Returns the pulse-pulse carrier phase in the pulse table.

Conditions

Measurement views: Pulse table

Group

Read commands

Syntax

`READ:PULSe[:RESult][:MEASview<y>]:PPPHase?`

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<first_pulse_num>,<pp_phase(1)>,<pp_phase(2)>,...<pp_phase(n)>`

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

`<pp_phase(n)>::<NRf>` is the pulse-pulse carrier phase for the pulse with the number of [first_pulse_num + n - 1] in degrees.

Examples

`READ:PULSe:RESult:MEASview2:PPPHase?` might return `0,0.000,21.66,46.76,57.56` for Pulse 0 to 3.

READ:PULSe[:RESult][:MEASview<y>]:RINTerval (Query Only)

Returns the repetition interval in the pulse table.

Conditions

Measurement views: Pulse table

Group

Read commands

Syntax

`READ:PULSe[:RESult][:MEASview<y>]:RINTerval?`

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<first_pulse_num>,<rep_int(1)>,<rep_int(2)>,...<rep_int(n)>`

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

`<rep_int(n)>::<NRf>` is the repetition interval for the pulse with the number of [first_pulse_num + n - 1] in seconds.

Examples

`READ:PULSe:RESult:MEASview2:RINTerval?` might return `0,16.03E-6,16.08E-6, 16.07E-6,16.02E-6` for Pulse 0 to 3.

READ:PULSe[:RESult][:MEASview<y>]:RIPDb (Query Only)

Returns the ripple in dB in the pulse table.

Conditions

Measurement views: Pulse table

Group

Read commands

Syntax

`READ:PULSe[:RESult][:MEASview<y>]:RIPDb?`

Related Commands

READ:PULSe[:RESult][:MEASview<y>]:RIPPle? (on page 1787)

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<first_pulse_num>,<ripple(1)>,<ripple(2)>,...<ripple(n)>`

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

`<ripple(n)>::<NRf>` is the ripple for the pulse with the number of [first_pulse_num + n - 1] in dB.

Examples

`READ:PULSe:RESult:MEASview2:RIPDb?` might return `1,0.4,0.32,0.2,0.1` for Pulse 1 to 4.

READ:PULSe[:RESult][:MEASview<y>]:RISE (Query Only)

Returns the rise time in the pulse table.

Conditions

Measurement views: Pulse table

Group

Read commands

Syntax

`READ:PULSe[:RESult][:MEASview<y>]:RISE?`

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<first_pulse_num>,<rise(1)>,<rise(2)>,...<rise(n)>`

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

`<rise(n)>::<NRf>` is the rise time for the pulse with the number of [first_pulse_num + n - 1] in seconds.

Examples

`READ:PULSe:RESult:MEASview2:RISE?` might return `0,92.94E-9,115.9E-9, 115.1E-9,97.45E-9` for Pulse 0 to 3.

READ:PULSe[:RESult][:MEASview<y>]:RIPPle (Query Only)

Returns the ripple in the pulse table.

Conditions

Measurement views: Pulse table

Group

Read commands

Syntax

`READ:PULSe[:RESult][:MEASview<y>]:RIPPle?`

Related Commands

READ:PULSe[:RESult][:MEASview<y>]:RIPDb? (on page 1786)

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<first_pulse_num>,<ripple(1)>,<ripple(2)>,...<ripple(n)>`

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

`<ripple(n)>::<NRf>` is the watt or voltage ripple for the pulse with the number of [first_pulse_num + n - 1] in percent (%).

Examples

`READ:PULSe:RESult:MEASview2:RIPPle?` might return `0,106.5E-3,177.6E-3, 148.3E-3,148.5E-3` for Pulse 0 to 3.

READ:PULSe[:RESult][:MEASview<y>]:RMSFreqerror (Query Only)

Returns the RMS frequency error in the pulse table.

Conditions

Measurement views: Pulse table

Group

Read commands

Syntax

`READ:PULSe[:RESult][:MEASview<y>]:RMSFreqerror?`

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<first_pulse_num>,<RMS_freq_err(1)>,<RMS_freq_err(2)>,...<RMS_freq_err(n)>`

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

`<RMS_freq_err(n)>::<NRf>` is the RMS frequency error for the pulse with the number of [first_pulse_num + n - 1] in Hz.

Examples

`READ:PULSe:RESult:MEASview2:RMSFreqerror?` might return `1,51.54E+3, 69.20E+3,64.21E+3,51.02E+3` for Pulse 1 to 4.

READ:PULSe[:RESult][:MEASview<y>]:RMSPherror (Query Only)

Returns the RMS phase error in the pulse table.

Conditions

Measurement views: Pulse table

Group

Read commands

Syntax

`READ:PULSe[:RESult][:MEASview<y>]:RMSPherror?`

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<first_pulse_num>,<RMS_phase_err(1)>,<RMS_phase_err(2)>,...<RMS_phase_err(n)>`

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

`<RMS_phase_err(n)>::<NRf>` is the RMS phase error for the pulse with the number of [first_pulse_num + n - 1] in degrees.

Examples

`READ:PULSe:RESult:MEASview2:RMSPherror?` might return `1,908.4E-3,752.8E-3,981.7E-3,886.4E-3` for Pulse 1 to 4.

READ:PULSe[:RESult][:MEASview<y>]:RRATe (Query Only)

Returns the repetition rate in the pulse table.

Conditions

Measurement views: Pulse table

Group

Read commands

Syntax

`READ:PULSe[:RESult][:MEASview<y>]:RRATe?`

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<first_pulse_num>,<rep_rate(1)>,<rep_rate(2)>,…<rep_rate(n)>`

Where

`<first_pulse_num>::<NR1>` is the first pulse number.

<rep_rate(n)>::<NRf> is the repetition rate for the pulse with the number of [first_pulse_num + n - 1] in Hz.

Examples

READ:PULSe:RESult:MEASview2:RRATe? might return 0,62.50E+3,62.52E+3, 62.51E+3,62.49E+3 for Pulse 0 to 3.

READ:PULSe[:RESult][:MEASview<y>]:TIME (Query Only)

Returns the time in the pulse table.

Conditions

Measurement views: Pulse table

Group

Read commands

Syntax

READ:PULSe[:RESult][:MEASview<y>]:TIME

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<first_pulse_num>,<time(1)>,<time(2)>,...<time(n)>

Where

<first_pulse_num>::<NR1> is the first pulse number.

<time(n)>::<NRf> is the time for the pulse with the number of [first_pulse_num + n - 1] in seconds.

Examples

READ:PULSe:RESult:MEASview2:TIME? might return 1,7.937E-3,8.436E-3, 6.504E-3,9.876E-3 for Pulse 1 to 4.

READ:PULSe[:RESult][:MEASview<y>]:WIDTh (Query Only)

Returns the pulse width in the pulse table.

Conditions

Measurement views: Pulse table

Group

Read commands

Syntax

READ:PULSe[:RESult][:MEASview<y>]:WIDTh?

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<first_pulse_num>,<width(1)>,<width(2)>,...<width(n)>

Where

<first_pulse_num>::<NR1> is the first pulse number.

<width(n)>::<NRf> is the pulse width for the pulse with the number of [first_pulse_num + n - 1] in seconds.

Examples

READ:PULSe:RESult:MEASview2:WIDTh? might return 0,4.630E-6,4.632E-6, 4.639E-6,4.642E-6 for Pulse 0 to 3.

READ:PULSe:STATistics[:MEASview<y>] (Query Only)

Returns the trace data of the pulse statistics measurement selected by the DISPlay:PULSe:SELect:RESult command.

NOTE: Select the plot type (Trend, FFT, Time Trend or Histogram) using the DISPlay:PULSe:STATistics:PLOT command before executing this query.

Conditions

Measurement views: Pulse statistics

Group

Read commands

Syntax

READ:PULSe:STATistics[:MEASview<y>]?

Arguments

None

Returns

#<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the amplitude at the nth data point.

The unit is current measurement unit (Plot = Trend or Time Trend), dB (Plot = FFT), count or percent (Plot = Histogram)

4-byte little endian floating-point format specified in IEEE 488.2.

The unit of power is selected by the [SENSe]:POWer:UNITs command.

Examples

READ:PULSe:STATistics:MEASview1? might return #264xxxx... (64-byte data) for the statistics trace of the pulse width measurement when DISPlay:PULSe:SELect:RESult is set to WIDTh.

READ:PULSe:STATistics[:MEASview<y>]:ADVanced (Query Only)

Returns the Pulse Count, Max, Max at Pulse number, Min, Min at Pulse number, Peak to Peak, Average and Standard Deviation values for the chosen measurement in pulse statistics display.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Read commands

Syntax

`READ:PULSe:STATistics[:MEASview<y>]:ADVanced? <arg1>,<arg2>`

Arguments

`arg1` can be one of the following measurements:

`{AVERage| PPOWer| ATX|WIDTh| RISE| FALL| RINTerval| RRATe|DUTPct| DUTRatio|RIPPle| RIPDb| DROop| DRODb| OVERshoot|OVEDb| PPPHase| PPFRequency| RMSFreqerror| MFReqerror|RMSPherror| MPHerror| FRDeviation| FDELta| PHDeviation|IRAMplitude|IRTime|PPPD|PPFD|FABS}`

`arg2` can be one of the following statistics results:

`{TPCOunt|MAX|MIN|PK2PK|AVG|SD|ALL}`

Returns

`<puls_cnt>|<max>|<min>|pk_pk>|<avg>|std_dev>|<all>` for one of the following selected measurements.

Where

`<avg_on>|<pk_pwr>|<avg_tx>|<width>|<rise>|<fall>|<rep_int>|<rep_rate>|<avg_on> | <pk_pwr> | <avg_tx> | <width> | <rise> | <fall> | <rep_int> | <rep_rate> | <duty_factor_pcnt> | <duty_fact_ratio> | <ripple> | <ripple_db> | <droop> | <droop_db> | <ovrshoot> | <ovrshoot_db> | <pp_phase> | <pp_freq> | <rms_phase_err> | <rms_freq_err> | <p2p_phase> | <p2p_freq> | <pk_phase_err> | <pk_freq_err> | <abs_freq> | <freq_dev> | <phase_dev> | <dlta_freq> | <imp_resp_amp> | <imp_resp_time>|<time>`

Where

`<avg_on>::<NRf>` is the average power transmitted during pulse on.

`<pk_pwr>::<NRf>` is the Maximum power during pulse on

`<avg_tx>:: <NRf>` is the average power transmitted

`<width>::<NRf>` is the time from the rising edge to the falling edge at the −3 dB /−6 dB level (50%) of the user selected 100% level

`<rise>::<NRf>` is the time required for a signal to rise from 10% to 90% (or 20% to 80%) of the user selected 100% level

`<fall> ::<NRf>` is the time required for a signal to fall from 90% to 10% (or 80% to 20%) of the user selected 100% level

`<rep_int>::<NRf>` is the time from a pulse rising edge to the next pulse rising edge

`<rep_rate> ::<NRf>` is the inverse of repetition interval

`<duty_factor_pcnt>::<NRf>` is the ratio of the width to the pulse period, expressed as a percentage

`<duty_fact_ratio>::<NRf>` is the ratio of the pulse width to the pulse period

`<ripple>::<NRf>` is the Ripple is the peak-to-peak ripple on the pulse top

<ripple_db>::<NRf> is the Ripple measurement expressed in dB

<droop>::<NRf> is the Droop is the power difference between the beginning and the end of the pulse On time

<droop_db>::<NRf> is the Droop measurement expressed in dB

<ovrshoot>::<NRf> is the amount by which the signal exceeds the 100% level on the pulse rising edge

<ovrshoot_db>::<NRf> is the Overshoot measurement expressed in dB

<pp_phase>::<NRf> is the phase difference between the selected pulse and the previous pulse in the analysis window

<pp_freq> ::<NRf> is the difference between the frequency of the current pulse and frequency of the previous pulse

<rms_phase_err>::<NRf> is the RMS average of the Phase vs Time trace, computed over the Measurement Time

<rms_freq_err>::<NRf> is the RMS average of the Freq Error vs.Time trace

<p2p_phase>::<NRf> is the frequency difference between the current pulse and the first pulse

<p2p_freq>::<NRf> is the frequency difference between the current pulse and immediate previous pulse

<pk_phase_err>::<NRf> is the maximum phase error

<abs_freq>::<NRf> is the absolute pulse frequency measured at a specific point

<freq_dev>::<NRf> is the difference between the maximum and minimum measured values of the signal frequency during the Measurement Time

<phase_dev>::<NRf> is the Phase Deviation difference between the maximum and minimum Phase values measured during the On time of a pulse

<dlta_freq>::<NRf> is the difference from the measurement frequency to each pulse frequency

<imp_resp_amp>::<NRf> is the difference in dB between the levels of the main lobe and highest side lobe

<imp_resp_time>::<NRf> is the difference in time between the main lobe and highest side lobe

<time>::<NRf> is the dtime in seconds relative to the time reference point in the first acquisition record in the data set.

Examples

`READ:PULSe:STATistics:MEASview2:ADVanced RISE, MAX` might return `7.9786177665E-9,"-"`, indicating the maximum rise value is 7.9786177665E-9.

READ:PULSe[:RESult]:ADVanced (Query Only)

Returns the PulseCount, Max, Max TimeStamp, Min, Min TimeStamp, Peak to Peak, Average, and Standard Deviation values for the chosen measurement in the Pulse Table display.

Conditions

Measurement views: All

Group

Read commands

Syntax

`READ:PULSe[:RESult]:ADVanced? <arg1><arg2>`

Arguments

<arg1> = {Pulse_number}

<arg2> = {AVERage | PPOWer | ATX | WIDTh | RISE | FALL | RINTerval | RRATe | DUTPct | DUTRatio |RIPPle | RIPDb | DROop | DRODb | OVERshoot | OVEDb | PPPHase | PPFRequency | RMSFreqerror | MFReqerror | RMSPherror | MPHerror | FRDeviation | FDELta | PHDeviation | IRAMplitude|TIME | IRTime | PPPD| PPFD | FABS}

Returns

<puls_cnt> | <max> | <max_timstmp> | <min> | <min_timstmp> | <pk_pk> | <avg> | <std_dev> | <all>

For one of the following selected measurements:

<avg_on> | <pk_pwr> | <avg_tx> | <width> | <rise> | <fall> | <rep_int> | <rep_rate> | <duty_factor_pcnt> | <duty_fact_ratio> | <ripple> | <ripple_db> | <droop> | <droop_db> | <ovrshoot> | <ovrshoot_db> | <pp_phase> | <pp_freq> | <rms_phase_err> | <rms_freq_err> | <p2p_phase> | <p2p_freq> | <pk_phase_err> | <pk_freq_err> | <abs_freq> | <freq_dev> | <phase_dev> | <dlta_freq> | <imp_resp_amp> | <imp_resp_time>|<time> for the given pulse number

where

<avg_on>::<NRf> is the average power transmitted during pulse on.

<pk_pwr>::<NRf> is the Maximum power during pulse on

<avg_tx>:: <NRf> is the average power transmitted

<width>::<NRf> is the time from the rising edge to the falling edge at the −3 dB / −6 dB level (50%) of the user selected 100% level

<rise>::<NRf> is the time required for a signal to rise from 10% to 90% (or 20% to 80%) of the user selected 100% level

<fall> ::<NRf> is the time required for a signal to fall from 90% to 10% (or 80% to 20%) of the user selected 100% level

<rep_int>::<NRf> is the time from a pulse rising edge to the next pulse rising edge

<rep_rate> ::<NRf> is the inverse of repetition interval

<duty_factor_pcnt>::<NRf> is the ratio of the width to the pulse period, expressed as a percentage

<duty_fact_ratio>::<NRf> is the ratio of the pulse width to the pulse period

<ripple>::<NRf> is the Ripple is the peak-to-peak ripple on the pulse top

<ripple_db>::<NRf> is the Ripple measurement expressed in dB

<droop>::<NRf> is the Droop is the power difference between the beginning and the end of the pulse On time

<droop_db>::<NRf> is the Droop measurement expressed in dB

<ovrshoot>::<NRf> is the amount by which the signal exceeds the 100% level on the pulse rising edge

<ovrshoot_db>::<NRf> is the Overshoot measurement expressed in dB

<pp_phase>::<NRf> is the phase difference between the selected pulse and the previous pulse in the analysis window

<pp_freq> ::<NRf> is the difference between the frequency of the current pulse and frequency of the previous pulse

<rms_phase_err>::<NRf> is the RMS average of the Phase vs Time trace, computed over the Measurement Time

<rms_freq_err>::<NRf> is the RMS average of the Freq Error vs.Time trace

<p2p_phase>::<NRf> is the frequency difference between the current pulse and the first pulse

<p2p_freq>::<NRf> is the frequency difference between the current pulse and immediate previous pulse

<pk_phase_err>::<NRf> is the maximum phase error

<abs_freq>::<NRf> is the absolute pulse frequency measured at a specific point

<freq_dev>::<NRf> is the difference between the maximum and minimum measured values of the signal frequency during the Measurement Time

<phase_dev>::<NRf> is the Phase Deviation difference between the maximum and minimum Phase values measured during the On time of a pulse

<dlta_freq>::<NRf> is the difference from the measurement frequency to each pulse frequency

<imp_resp_amp>::<NRf> is the difference in dB between the levels of the main lobe and highest side lobe

<imp_resp_time>::<NRf> is the difference in time between the main lobe and highest side lobe

<time>::<NRf> is the dtime in seconds relative to the time reference point in the first acquisition record in the data set

Examples

READ:PULSE[:RESULT]:ADVANCED 1,AVERage might return -1.4063608208, indicating the average ON value for the pulse number 1 is −1.4063608208.

READ:PULSe:STATistics[:MEASview<y>]:AVERage (Query Only)

Returns the average on power in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT is set to TRENd.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Read commands

Syntax

READ:PULSe:STATistics[:MEASview<y>]:AVERage

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<avg_avg>,<avg_min>,< avg_max>

Where

<avg_avg>::<NRf> is the average of the average on power.

<avg_min>::<NRf> is the minimum of the average on power.

<avg_max>::<NRf> is the maximum of the average on power.

Unit: dBm.

The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples

READ:PULSe:STATistics:MEASview2:AVERage might return -2.35,-2.36,-2.34 for the average on power in the pulse statistics.

READ:PULSe:STATistics[:MEASview<y>]:DRODb (Query Only)

Returns the droop in dB in the pulse statistics. This command is valid when `DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT` is set to `TRENd`.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Read commands

Syntax

`READ:PULSe:STATistics[:MEASview<y>]:DRODb`

Related Commands

`READ:PULSe:STATistics[:MEASview<y>]:DROop?`

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<droop_avg>,<droop_min>,<droop_max>`

Where

`<droop_avg>::<NRf>` is the average droop.

`<droop_min>::<NRf>` is the minimum droop.

`<droop_max>::<NRf>` is the maximum droop.

Unit: in dB by watts.

Examples

`READ:PULSe:STATistics:MEASview2:DRODb?` might return `22.67E-3,-613.5E-3, 633.8E-3` for the droop in the pulse statistics measurement.

READ:PULSe:STATistics[:MEASview<y>]:DROop (Query Only)

Returns the droop in the pulse statistics.This command is valid when `DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT` is set to `TRENd`.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Read commands

Syntax

`READ:PULSe:STATistics[:MEASview<y>]:DROop`

Related Commands

READ:PULSe:STATistics[:MEASview<y>]:DRODb? (on page 1796)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<droop_avg>,<droop_min>,<droop_max>`

Where

`<droop_avg>::<NRf>` is the average droop.

`<droop_min>::<NRf>` is the minimum droop.

`<droop_max>::<NRf>` is the maximum droop.

Unit: Percent (%) by watts.

Examples

`READ:PULSe:STATistics:MEASview2:DROop` might return `22.67E-3,-613.5E-3, 633.8E-3` for the droop in the pulse statistics measurement.

READ:PULSe:STATistics[:MEASview<y>]:DUTPct (Query Only)

Returns the duty factor (%) in the pulse statistics. This command is valid when `DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT` is set to `TRENd`.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Read commands

Syntax

`READ:PULSe:STATistics[:MEASview<y>]:DUTPct`

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<duty_pct_avg>,<duty_pct_min)>,<duty_pct_max>`

Where

`<duty_pct_avg>::<NRf>` is the average duty factor.

`<duty_pct_min>::<NRf>` is the minimum duty factor.

`<duty_pct_max>::<NRf>` is the maximum duty factor.

Unit: Percent (%).

Examples

`READ:PULSe:STATistics:MEASview2:DUTPct` might return `2.437,2.310,2.657` for the duty factor (%) in the pulse statistics measurement.

READ:PULSe:STATistics[:MEASview<y>]:DUTRatio (Query Only)

Returns the duty factor (ratio) in the pulse statistics. This command is valid when `DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT` is set to `TRENd`.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Read commands

Syntax

`READ:PULSe:STATistics[:MEASview<y>]:DUTRatio`

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<duty_ratio_avg>,<duty_ratio_min>,<duty_ratio_max)>`

Where

`<duty_ratio_avg>::<NRf>` is the average duty factor.

`<duty_ratio_min>::<NRf>` is the minimum duty factor.

`<duty_ratio_max>::<NRf>` is the maximum duty factor.

Unit: None.

Examples

`READ:PULSe:STATistics:MEASview2:DUTRatio?` might return `24.37E-3,23.11E-3, 26.57E-3` for the duty factor (ratio) in the pulse statistics measurement.

READ:PULSe:STATistics[:MEASview<y>]:FABS (Query Only)

Returns the absolute frequency in pulse statistics.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Read commands

Syntax

`READ:PULSe:STATistics[:MEASview<y>]:FABS`

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<fabs_avg>,<fabs_min>,<fabs_max>`

Where

<fabs_avg>::<NRf> is the average of fabs.

<fabs_min>::<NRf> is the minimum of fabs.

<fabs_max>::<NRf> is the maximum of fabs.

Examples

READ:PULSe:STATistics:MEASview2:FABS might return 1.1196544000E+9,919.8726400000E+6,1.1929450240E+9 for the absolute frequency in the pulse statistics.

READ:PULSe:STATistics[:MEASview<y>]:FALL (Query Only)

Returns the fall time in the pulse statistics. .This command is valid when DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT is set to TRENd.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Read commands

Syntax

READ:PULSe:STATistics[:MEASview<y>]:FALL

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<fall_avg>,<fall_min>,<fall_max>

Where

<fall_avg>::<NRf> is the average fall time.

<fall_min>::<NRf> is the minimum fall time.

<fall_max>::<NRf> is the maximum fall time.

Unit: Seconds.

Examples

READ:PULSe:STATistics:MEASview2:FALL? might return 70.27E-9,69.62E-9, 71.27E-9 for the fall time in the pulse statistics.

READ:PULSe:STATistics[:MEASview<y>]:FDELta (Query Only)

Return the delta frequency measurement in the pulse statistics.

Group

Read commands

Syntax

READ:PULSe:STATistics[:MEASview<y>]:FDELta

Arguments

None

Returns

<freq_delta_avg>,<freq_delta_min>,<freq_delta_max>

Where

<freq_delta_avg>::<NRf> is the average frequency deviation.

<freq_delta_min>::<NRf> is the minimum frequency deviation.

<freq_delta_max>::<NRf> is the maximum frequency deviation.

Unit: Hz.

Examples

READ:PULSe:STATistics:MEASview2:FDELta? might return 77.9785690308,-110.1940994263,287.8957519531 for the frequency delta in the pulse statistics.

READ:PULSe:STATistics[:MEASview<y>]:FRDeviation (Query Only)

Returns the frequency deviation in the Pulse Statistics measurement. This command is valid when DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT is set to TRENd.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Read commands

Syntax

READ:PULSe:STATistics[:MEASview<y>]:FRDeviation

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<freq_dev_avg>,<freq_dev_min>,<freq_dev_max>

Where

<freq_dev_avg>::<NRf> is the average frequency deviation.

<freq_dev_min>::<NRf> is the minimum frequency deviation.

<freq_dev_max>::<NRf> is the maximum frequency deviation.

Unit: Hz.

Examples

READ:PULSe:STATistics:MEASview2:FRDeviation? might return 754.1E+3, 660.5E+3,835.2E+3 for the frequency deviation in the pulse statistics measurement.

READ:PULSe:STATistics[:MEASview<y>]:IRAMplitude (Query Only)

Returns the Pulse Impulse Response amplitude measurement in the pulse statistics.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Read commands

Syntax

`READ:PULSe:STATistics[:MEASview<y>]:IRAMplitude`

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<amp_avg>,<amp_min>,<amp_max>`

Where

`<amp_avg>::<NRf>` is the average of the impulse response amplitude.

`<amp_min>::<NRf>` is the minimum of the impulse response amplitude.

`<amp_max>::<NRf>` is the maximum of impulse response amplitude.

Unit: dB.

Examples

`READ:PULSe:STATistics:MEASview2:IRAMplitude?` might return `-45.3,-47.1,-43.2` for the impulse response amplitude in the pulse statistics.

READ:PULSe:STATistics[:MEASview<y>]:IRTime (Query Only)

Returns the impulse response time in the pulse statistics.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Read commands

Syntax

`READ:PULSe:STATistics[:MEASview<y>]:IRTime`

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<avg_avg>,<avg_min>,<avg_max>`

Where

<avg_avg>::<NRf> is the average of the impulse response time.

<avg_min>::<NRf> is the minimum of the impulse response time.

<avg_max>::<NRf> is the maximum of impulse response time.

Unit: seconds.

Examples

READ:PULSe:STATistics:MEASview2:IRTime? might return -2.35,-2.36,-2.34 for the impulse response time in the pulse statistics.

READ:PULSe:STATistics[:MEASview<y>]:MAX:PULSe:NUMber (Query Only)

Returns the maximum value at a certain pulse number in the pulse statistics display.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Read commands

Syntax

READ:PULSe:STATistics[:MEASview<y>]:MAX:PULSe:NUMber

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<max_pulse_no>

Where

<max_pulse_no>::<NRf> is the maximum value at a pulse number.

Examples

READ:PULSe:STATistics:MEASview2:MAX:PULSe:NUMber? might return 15 for the maximum pulse number in the pulse statistics.

READ:PULSe:STATistics[:MEASview<y>]:MFReqerror (Query Only)

Returns the maximum frequency error in the pulse statistics measurement. This command is valid when DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT is set to TRENd.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Read commands

Syntax

`READ:PULSe:STATistics[:MEASview<y>]:MFReqerror`

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<max_freq_err_avg>,<max_freq_err_min>,<max_freq_err_max>`

Where

`<max_freq_err_avg>::<NRf>` is the average of maximum frequency error.

`<max_freq_err_min>::<NRf>` is the minimum of maximum frequency error.

`<max_freq_err_max>::<NRf>` is the maximum of maximum frequency error.

Unit: Hz.

Examples

`READ:PULSe:STATistics:MEASview2:MFReqerror?` might return `645.0E+3,555.6E+3,738.8E+3` for the maximum frequency error in the pulse statistics measurement.

READ:PULSe:STATistics[:MEASview<y>]:MIN:PULSe:NUMber? (Query Only)

Returns the minimum value at a certain pulse number in pulse statistics.

Conditions

Measurement views: Pulse statistics

Group

Read commands

Syntax

`READ:PULSe:STATistics[:MEASview<y>]:MIN:PULSe:NUMber?`

Arguments

None

Returns

`<min_pulse_no>`

Where

`<min_pulse_no>::<NRf>` is the minimum value at a pulse number.

Examples

`READ:PULSE:STATISTICS:MEASview1:MIN:PULSE:NUMBER?` might return `1` for the minimum pulse number in the pulse statistics

READ:PULSe:STATistics[:MEASview<y>]:MPHerror (Query Only)

Returns the maximum phase error in the Pulse Statistics measurement . This command is valid when `DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT` is set to `TRENd`.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Read commands

Syntax

`READ:PULSe:STATistics[:MEASview<y>]:MPHerror`

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<max_phase_err_avg>,<max_phase_err_min>,<max_phase_err_max>`

Where

`<max_phase_err_avg>::<NRf>` is the average of maximum phase error.

`<max_phase_err_min>::<NRf>` is the minimum of maximum phase error.

`<max_phase_err_max>::<NRf>` is the maximum of maximum phase error.

Unit: Degrees.

Examples

`READ:PULSe:STATistics:MEASview2:MPHerror?` might return `-11.47,-17.18,-7.61` for the maximum phase error in the pulse statistics measurement.

READ:PULSe:STATistics[:MEASview<y>]:OVEDb (Query Only)

Returns the overshoot measurement result in dB in the pulse statistics. This command is valid when `DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT` is set to `TRENd` or `TTRend`.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Read commands

Syntax

`READ:PULSe:STATistics[:MEASview<y>]:OVEDb`

Related Commands

READ:PULSe:STATistics[:MEASview<y>]:OVERshoot (on page 1805)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<overshoot_avg>,<overshoot_min>,<overshoot_max>

Where

<overshoot_avg>::<NRf> is the average overshoot.

<overshoot_min>::<NRf> is the minimum overshoot.

<overshoot_max>::<NRf> is the maximum overshoot.

Unit: dB.

Examples

READ:PULSe:STATistics:MEASview2:OVEDb might return 0.3,0.1,0.5 for the overshoot measurement result in dB in the pulse statistics measurement.

READ:PULSe:STATistics[:MEASview<y>]:OVERshoot (Query Only)

Returns the overshoot in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT is set to TRENd or TTRend.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Read commands

Syntax

READ:PULSe:STATistics[:MEASview<y>]:OVERshoot

Related Commands

READ:PULSe:STATistics[:MEASview<y>]:OVEDb (on page 1804)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<overshoot_avg>,<overshoot_min>,<overshoot_max>

Where

<overshoot_avg>::<NRf> is the average overshoot.

<overshoot_min>::<NRf> is the minimum overshoot.

<overshoot_max>::<NRf> is the maximum overshoot.

Unit: Percent (%) as watts or volts.

Examples

READ:PULSe:STATistics:MEASview2:OVERshoot might return 0.3,0.1,0.5 for the overshoot in the pulse statistics measurement.

READ:PULSe:STATistics[:MEASview<y>]:PHDeviation (Query Only)

Returns the phase deviation in the Pulse Statistics measurement. This command is valid when `DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT` is set to `TRENd`

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Read commands

Syntax

`READ:PULSe:STATistics[:MEASview<y>]:PHDeviation`

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<phase_dev_avg>,<phase_dev_min>,<phase_dev_max>`

Where

`<phase_dev_avg>::<NRf>` is the average phase deviation.

`<phase_dev_min>::<NRf>` is the minimum phase deviation.

`<phase_dev_max>::<NRf>` is the maximum phase deviation.

Unit: Degrees.

Examples

`READ:PULSe:STATistics:MEASview2:PHDeviation` might return `11.678,7.694, 17.374` for the phase deviation in the pulse statistics measurement.

READ:PULSe:STATistics[:MEASview<y>]:PPFD (Query Only)

Returns the pulse-pulse frequency difference in the Pulse Statistics measurement display.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Read commands

Syntax

`READ:PULSe:STATistics[:MEASview<y>]:PPFD`

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<ppfd_avg>,<ppfd_min>,<ppfd_max>`

Where

<ppfd_avg>::<NRf> is the average of ppfd.

<ppfd_min>::<NRf> is the minimum of ppfd.

<ppfd_max>::<NRf> is the maximum of ppfd.

Examples

READ:PULSe:STATistics:MEASview2:PPFD? might return 365.4186562500E+3,-267.5911040000E+6,247.1117760000E+6 for the pulse-pulse frequency difference in the pulse statistics.

READ:PULSe:STATistics[:MEASview<y>]:PPFRequency (Query Only)

Returns the pulse-pulse carrier frequency in the pulse statistics measurement. This command is valid when DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT is set to TRENd.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Read commands

Syntax

READ:PULSe:STATistics[:MEASview<y>]:PPFRequency

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<pp_freq_avg>,<pp_freq_min)>,<pp_freq_max>

Where

<pp_freq_avg>::<NRf> is the average pulse-pulse carrier frequency.

<pp_freq_min>::<NRf> is the minimum pulse-pulse carrier frequency.

<pp_freq_max>::<NRf> is the maximum pulse-pulse carrier frequency.

Unit: Hz.

Examples

READ:PULSe:STATistics:MEASview2:PPFRequency? might return 1.135E+3,311.3E+3,-262.8E+3 for the pulse-pulse carrier frequency in the pulse statistics measurement.

READ:PULSe:STATistics[:MEASview<y>]:PPOWer (Query Only)

Returns the peak power in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT is set to TRENd.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Read commands

Syntax

`READ:PULSe:STATistics[:MEASview<y>]:PPOWer`

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<pk_power_avg>,<pk_power_min>,<pk_power_max>`

Where

`<pk_power_avg>::<NRf>` is the average peak power.

`<pk_power_min>::<NRf>` is the minimum peak power.

`<pk_power_max>::<NRf>` is the maximum peak power.

Unit: dBm.

The unit can be changed by the `[SENSe]:POWer:UNITs` command.

Examples

`READ:PULSe:STATistics:MEASview2:PPOWer?` might return `-2.273,-2.313,-2.235` for the peak power in the pulse statistics measurement.

READ:PULSe:STATistics[:MEASview<y>]:PPPD (Query Only)

Returns the overshoot measurement result in dB in the pulse statistics measurement. This command is valid when `DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT` is set to `TRENd` or TTRend.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Read commands

Syntax

`READ:PULSe:STATistics[:MEASview<y>]:PPPD`

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<pppd_avg>,<pppd_min>,<pppd_max>`

Where

`<pppd_avg>::<NRf>` is the average of pppd.

`<pppd_min>::<NRf>` is the minimum of pppd.

`<pppd_max>::<NRf>` is the maximum of pppd.

Examples

`READ:PULSe:STATistics:MEASview2:PPPD?` might return `-1.1131216288,-151.8063964844,130.5854492188`, indicating the pulse-pulse phase difference in dB.

READ:PULSe:STATistics[:MEASview<y>]:PPPHase (Query Only)

Returns the overshoot measurement result in dB in the pulse statistics measurement. This command is valid when `DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT` is set to `TRENd` or TTRend.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Read commands

Syntax

`READ:PULSe:STATistics[:MEASview<y>]:PPPHase`

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<pp_phase_avg>,<pp_phase_min>,<pp_phase_max>`

Where

`<pp_phase_avg>::<NRf>` is the average pulse-pulse carrier phase.

`<pp_phase_min>::<NRf>` is the minimum pulse-pulse carrier phase.

`<pp_phase_max>::<NRf>` is the maximum pulse-pulse carrier phase.

Unit: Degrees.

Examples

`READ:PULSe:STATistics:MEASview2:PPPHase?` might return `-9.298E-3,-254.3E-3,311.7E-3` for the pulse-pulse carrier phase in the pulse statistics measurement.

READ:PULSe:STATistics[:MEASview<y>]:RINTerval (Query Only)

Returns the repetition interval in the pulse statistics. This command is valid when `DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT` is set to `TRENd`.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Read commands

Syntax

`READ:PULSe:STATistics[:MEASview<y>]:RINTerval`

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<rep_int_avg>,<rep_int_min>,<rep_int_max>

Where

<rep_int_avg>::<NRf> is the average repetition interval.

<rep_int_min>::<NRf> is the minimum repetition interval.

<rep_int_max>::<NRf> is the maximum repetition interval.

Unit: Seconds.

Examples

READ:PULSe:STATistics:MEASview2:RINTerval? might return 240.5E-6,217.9E-6,281.2E-6 for the repetition interval in the pulse statistics measurement.

READ:PULSe:STATistics[:MEASview<y>]:RIPDb (Query Only)

Returns the ripple in dB in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT is set to TRENd.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Read commands

Syntax

READ:PULSe:STATistics[:MEASview]:RIPDb

Related Commands

READ:PULSe:STATistics[:MEASview]:RIPPle (on page 1811)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<ripple_avg>,<ripple_min>,<ripple_max>

Where

<ripple_avg>::<NRf> is the average ripple.

<ripple_min>::<NRf> is the minimum ripple.

<ripple_max>::<NRf> is the maximum ripple.

Unit: dB.

Examples

READ:PULSe:STATistics:MEASview2:RIPDb? might return 160.4E-3,83.78E-3,287.7E-3 for the ripple in dB in the pulse statistics measurement.

READ:PULSe:STATistics[:MEASview<y>]:RIPPle (Query Only)

Returns the ripple in the pulse statistics. This command is valid when `DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT` is set to `TRENd`.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Read commands

Syntax

`READ:PULSe:STATistics[:MEASview]:RIPPle`

Related Commands

READ:PULSe:STATistics[:MEASview]:RIPDb (on page 1810)

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<ripple_avg>,<ripple_min>,<ripple_max>`

Where

`<ripple_avg>::<NRf>` is the average ripple.

`<ripple_min>::<NRf>` is the minimum ripple.

`<ripple_max>::<NRf>` is the maximum ripple.

Unit: Percent (%) as watts or volts.

Examples

`READ:PULSe:STATistics:MEASview2:RIPPle?` might return `60.4E-3,83.78E-3,287.7E-3` for the ripple in the pulse statistics measurement.

READ:PULSe:STATistics[:MEASview<y>]:RISE (Query Only)

Returns the rise time in the pulse statistics. This command is valid when `DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT` is set to `TRENd`.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Read commands

Syntax

`READ:PULSe:STATistics[:MEASview<y>]:RISE`

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<rise_avg>,<rise_min>,<rise_max>

Where

<rise_avg>::<NRf> is the average rise time.

<rise_min>::<NRf> is the minimum rise time.

<rise_max>::<NRf> is the maximum rise time.

Unit: Seconds.

Examples

READ:PULSe:STATistics:MEASview2:RISE? might return 105.4E-9,91.65E-9, 116.2E-9 for the rise time in the pulse statistics measurement.

READ:PULSe:STATistics[:MEASview<y>]:RMSFreqerror (Query Only)

Returns the RMS frequency error in the pulse statistics measurement. This command is valid when DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT is set to TRENd.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Read commands

Syntax

READ:PULSe:STATistics[:MEASview<y>]:RMSFreqerror

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<RMS_freq_err_avg>,<RMS_freq_err_min>,<RMS_freq_err_max>

Where

<RMS_freq_err_avg>::<NRf> is the average of RMS frequency error.

<RMS_freq_err_min>::<NRf> is the minimum of RMS frequency error.

<RMS_freq_err_max>::<NRf> is the maximum of RMS frequency error.

Unit: Hz.

Examples

READ:PULSe:STATistics:MEASview2:RMSFreqerror? might return 63.67E+3,45.49E+3,81.28E+3 for the RMS frequency error in the pulse statistics measurement.

READ:PULSe:STATistics[:MEASview<y>]:RMSPherror (Query Only)

Returns the RMS phase error in the Pulse Statistics measurement. This command is valid when `DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT` is set to `TRENd`.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Read commands

Syntax

`READ:PULSe:STATistics[:MEASview<y>]:RMSPherror`

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<RMS_phase_err_avg>,<RMS_phase_err_min>,<RMS_phase_err_max>`

Where

`<RMS_phase_err_avg>::<NRf>` is the average of RMS phase error.

`<RMS_phase_err_min>::<NRf>` is the minimum of RMS phase error.

`<RMS_phase_err_max>::<NRf>` is the maximum of RMS phase error.

Unit: Degrees.

Examples

`READ:PULSe:STATistics:MEASview2:RMSPherror?` might return `1.032,604.5E-3, 1.606` for the RMS phase error in the pulse statistics measurement.

READ:PULSe:STATistics[:MEASview<y>]:RRATe (Query Only)

Returns the repetition rate in the pulse statistics. This command is valid when `DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT` is set to `TRENd`.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Read commands

Syntax

`READ:PULSe:STATistics[:MEASview<y>]:RRATe`

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<rep_rate_avg>,<rep_rate_min>,<rep_rate_max>

Where

<rep_rate_avg>::<NRf> is the average repetition rate.

<rep_rate_min>::<NRf> is the minimum repetition rate.

<rep_rate_max>::<NRf> is the maximum repetition rate.

Unit: Hz.

Examples

READ:PULSe:STATistics:MEASview2:RRATe? might return 62.50E+3,62.49E+3,62.52E+3 for the repetition rate in the pulse statistics measurement.

READ:PULSe:STATistics[:MEASview<y>]:WIDTh (Query Only)

Returns the pulse width in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT is set to TRENd.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Read commands

Syntax

READ:PULSe:STATistics[:MEASview<y>]:WIDTh

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<width_avg>,<width_min>,<width_max>

Where

<width_avg>::<NRf> is the average pulse width.

<width_min>::<NRf> is the minimum pulse width.

<width_max>::<NRf> is the maximum pulse width.

Unit: Seconds.

Examples

READ:PULSe:STATistics:MEASview2:WIDTh? might return 4.636E-6,4.630E-6,4.643E-6 for the pulse width in the pulse statistics.

READ:PULSe:STATistics[:MEASview<y>]:X (Query Only)

Returns the horizontal values of the trace data of the pulse statistics measurement selected by the DISPlay:PULSe:SELect:RESult command.

NOTE: Select the plot type (Trend, FFT, Time Trend or Histogram) using the `DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT` command before executing this query.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Read commands

Syntax

`READ:PULSe:STATistics[:MEASview<y>]:X`

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`#<num_digit><num_byte><data1><data2>....data<n>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the horizontal value of the statistics graph at the nth data point.

The unit is trace number (Plot = Trend), Hz (Plot = FFT), second (Plot = Time Trend) or current measurement unit (Plot = Histogram).

4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`READ:PULSe:STATistics:MEASview2:X?` might return `#264xxxx...` (64-byte data) for the statistics trace of the pulse width measurement when `DISPlay:PULSe:SELect:RESult` is set to WIDTh.

READ:PULSe:STATistics[:MEASview<y>]:XY (Query Only)

Returns the horizontal and vertical values of the trace data of the pulse statistics measurement selected by the **DISPlay:PULSe:SELect:RESult** command.

NOTE: Select the plot type (Trend, FFT, Time Trend or Histogram) using the `DISPlay:PULSe:STATistics[:MEASview<y>]:PLOT` command before executing this query.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number

Group

Read commands

Syntax

`READ:PULSe:STATistics[:MEASview<y>]:XY`

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

#<num_digit><num_byte><data1><data2>....data<n>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the horizontal value of the statistics graph at the nth data point.

<data(n+1)> is the vertical value of the statistics graph at the nth data point.

For horizontal values, the unit is trace number (Plot = Trend), Hz (Plot = FFT), second (Plot = Time Trend) or current measurement unit (Plot = Histogram).

For vertical values, the unit is current measurement unit (Plot = Trend or Time Trend), dB (Plot = FFT), count (Plot = Histogram)

4-byte little endian floating-point format specified in IEEE 488.2.

Examples

READ:PULSe:STATistics:MEASview2:XY? might return #264xxxx... (64-byte data) for the statistics trace of the pulse width measurement when DISPlay:PULSe:SELect:RESult is set to WIDTh.

READ:PULSe:TRACe[:MEASview<y>]:Y

Returns the vertical values of the pulse trace. Use the DISPlay:PULSe:SELect:NUMBer command to select the pulse, and the DISPlay:PULSe:SELect:RESult command to select the measurement result.

Conditions

Measurement views: Pulse trace with specified Measurement view number

Group

Read commands

Syntax

READ:PULSe:TRACe[:MEASview<y>]:Y?

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

#<num_digit><num_byte><y(1)><y(2)>...<y(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<y(n)> is the vertical value of the pulse trace at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

The unit depends on measurement results: Hz for frequency error and deviation, degrees for phase error and deviation, otherwise dBm. The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples

READ:PULSE:TRACE:MEASVIEW1:Y? might return #43204xxxx... (3204-byte data) for the vertical values of the pulse trace measurement view 1.

READ:PULSe:TRACe[:MEASview<y>]:X

Returns the time values of the pulse trace. Use the DISPlay:PULSe:SELect:NUMBer command to select the pulse, and the DISPlay:PULSe:SELect:RESult command to select the measurement result.

Conditions

Measurement views: Pulse trace with specified Measurement view number

Group

Read commands

Syntax

READ:PULSe:TRACe[:MEASview<y>]:X?

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

#<num_digit><num_byte><X(1)><X(2)>...<X(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<X(n)> is the time in seconds at the nth data point,

4-byte little endian floating-point format specified in IEEE 488.2.

Examples

READ:PULSE:TRACE:MEASVIEW1:X? might return #43204xxxx... (3204-byte data) for the time values of the trace in pulse trace measurement view 1.

READ:PULSe:TRACe[:MEASview<y>]:XY

Returns the horizontal (time) and vertical value pairs of the pulse trace. Use the DISPlay:PULSe:SELect:NUMBer command to select the pulse, and the DISPlay:PULSe:SELect:RESult command to select the measurement result.

Conditions

Measurement views: Pulse trace with specified Measurement view number

Group

Read commands

Syntax

READ:PULSe:TRACe[:MEASview<y>]:XY?

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

#<num_digit><num_byte><x(1)><y(1)><x(2)><y(2)>...<x(n)><y(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<x(n)><y(n)> is the horizontal value (time in seconds) and vertical value pair at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

The vertical unit depends on measurement results: Hz for frequency error and deviation, degrees for phase error and deviation, otherwise dBm. The vertical unit can be changed by the [SENSe]:POWer:UNITs command.

Examples

READ:PULSE:TRACE:MEASVIEW1:XY? might return #43204xxxx... (3204-byte data) for the horizontal (time) and vertical value pairs of the pulse trace measurement view 1.

READ:SEM:MHITs:COUNt? (Query Only)

Returns the number of rows in the results table in the Spectral Emissions Mask measurement.

Conditions

Measurement views: Spectral Emissions Mask

Group

Read commands

Syntax

READ:SEM:MHITs:COUNt?

Arguments

None

Returns

<NRf> The number of rows in the table.

Examples

READ:SEM:MHITS:COUNT? might return 7, indicating there are seven rows in the results table.

READ:SEM:MHITs<x>:AMPLitude:ABSolute? (Query Only)

Returns the absolute amplitude of the specified mask hit signal in the Spectral Emissions Mask measurement.

Conditions

Measurement views: Spectral Emissions Mask

Group

Read commands

Syntax

`READ:SEM:MHITs<x>:AMPLitude:ABSolute?`

Arguments

None

Returns

`<NRf>` Absolute amplitude of the specified signal in dBm.

`<x>` specifies the row in the results table, where the range is from 1 to 12.

Examples

`READ:SEM:MHITs2:AMPLitude:ABSolute?` might return `-73.17` indicating that the absolute amplitude of signal in row 2 is –73.17 dBm.

READ:SEM:MHITs<x>:AMPLitude:RELative? (Query Only)

Returns the relative amplitude of the specified mask hit signal in the Spectral Emissions Mask measurement.

Conditions

Measurement views: Spectral Emissions Mask

Group

Read commands

Syntax

`READ:SEM:MHITs<x>:AMPLitude:RELative?`

Arguments

None

Returns

`<NRf>` Relative amplitude of the specified signal in dBm.

`<x>` specifies the row in the results table, where the range is from 1 to 12.

Examples

`READ:SEM:MHITs2:AMPLitude:RELative?` might return `-91.81` indicating that the relative amplitude of mask hit signal in row 2 is –91.81 dBm.

READ:SEM:MHITs<x>:FREQuency? (Query Only)

Returns the frequency of the specified mask hit signal in the Spectral Emissions Mask measurement.

Conditions

Measurement views: Spectral Emissions Mask

Group

Read commands

Syntax

READ:SEM:MHITs<x>:FREQuency?

Arguments

None

Returns

<NRf> Frequency of the specified signal in Hz.

<x> specifies the row in the results table, where the range is from 1 to 12.

Examples

READ:SEM:MHITs4:FREQuency? might return 2.235E+6, indicating the frequency for mask hit signal in row 4 is 2.235 MHz.

READ:SEM:MHITs<x>:INTeg:ABSolute? (Query Only)

Returns the absolute integration amplitude of the specified mask hit signal in the Spectral Emissions Mask measurement.

Conditions

Measurement views: Spectral Emissions Mask

Group

Read commands

Syntax

READ:SEM:MHITs<x>:INTeg:ABSolute?

Arguments

None

Returns

<NRf> Absolute integration amplitude of the specified signal in dBm.

<x> specifies the row in the results table, where the range is from 1 to 12.

Examples

READ:SEM:MHITs4:INTeg:ABSolute? might return −75.14 indicating that the absolute integration amplitude for mask hit signal in row 4 is −75.14 dBM.

READ:SEM:MHITs<x>:INTeg:RELative? (Query Only)

Returns the relative integration amplitude of the specified mask hit signal in the Spectral Emissions Mask measurement.

Conditions

Measurement views: Spectral Emissions Mask

Group

Read commands

Syntax

READ:SEM:MHITs<x>:INTeg:RELative?

Arguments

None

Returns

<NRf> Relative integration amplitude of the specified signal in dBm.

<x> specifies the row in the results table, where the range is from 1 to 12.

Examples

READ:SEM:MHITs4:INTeg:RELative? might return -85.14 indicating that the relative integration amplitude for mask hit signal in row 4 is –85.14 dBM.

READ:SEM:MHITs<x>:MARGin:ABSolute? (Query Only)

Returns the absolute margin for the specified mask hit signal in the Spectral Emissions Mask measurement.

Conditions

Measurement views: Spectral Emissions Mask

Group

Read commands

Syntax

READ:SEM:MHITs<x>:MARGin:ABSolute?

Arguments

None

Returns

<NRf> Absolute margin of the specified signal in dB.

<x> specifies the row in the results table, where the range is from 1 to 12.

Examples

READ:SEM:MHITs4:MARGin:ABSolute? might return -62.33 indicating that the absolute margin for mask hit signal in row 4 is –62.33 dB.

READ:SEM:MHITs<x>:MARGin:RELative? (Query Only)

Returns the relative margin for the specified mask hit signal in the Spectral Emissions Mask measurement.

Conditions

Measurement views: Spectral Emissions Mask

Group

Read commands

Syntax

READ:SEM:MHITs<x>:MARGin:RELative?

Arguments

None

Returns

<NRf> Relative margin of the specified signal in dB.

<x> specifies the row in the results table, where the range is from 1 to 12.

Examples

READ:SEM:MHITs4:MARGin:RELative? might return -62.33 indicating that the relative margin for mask hit signal in row 4 is –62.33 dB.

READ:SEM:MHITs<x>:OFFSet? (Query Only)

Returns the offset for the specified mask hit signal in the Spectral Emissions Mask measurement.

Conditions

Measurement views: Spectral Emissions Mask

Group

Read commands

Syntax

READ:SEM:MHITs<x>:OFFSet?

Arguments

None

Returns

<NRf> specifies the mask hit offset.

<x> specifies the row in the results table, where the range is from 1 to 12.

Examples

READ:SEM:MHITs4:OFFSet? might return BL, indicating the offset for mask hit signal in row 4 is BL.

READ:SEM:PASS? (Query Only)

Returns the pass/fail limit test result in the Spectral Emissions Mask measurement.

Conditions

Measurement views: Spectral Emissions Mask

Group

Read commands

Syntax

READ:SEM:PASS?

Arguments

None

Returns

0 (fail) or 1 (pass)

Examples

`READ:SEM:PASS?` might return `1`, indicating that the limit test was successful.

READ:SEM:REF:POWer? (Query Only)

Returns the reference power level in the Spectral Emissions Mask measurement.

Conditions

Measurement views: Spectral Emissions Mask

Group

Read commands

Syntax

`READ:SEM:REF:POWer?`

Arguments

None

Returns

`<NRf>` specifies the reference power.

Examples

`READ:SEM:REF:POWER?` might return `-16.92 dBm` indicating that the reference power level is –16.92 dBm.

READ:SEM:SPECtrum:X? (Query Only)

Returns the frequencies of the spectrum trace in the Spectral Emissions Mask measurement.

Conditions

Measurement views: Spectral Emissions Mask

Group

Read commands

Syntax

`READ:SEM:SPECtrum:X?`

Arguments

None

Returns

`#<num_digit><num_byte><x(1)><x(2)>...<x(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

<num_byte> is the number of bytes of data that follow.

<x(n)> is the frequency (Hz) at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

READ:SEM:SPECTRUM:X? might return #516020xxxx... (16020-byte data) for the frequencies of the spectrum trace.

READ:SEM:SPECtrum:XY? (Query Only)

Returns the frequency and amplitude pairs of the spectrum trace in the Spectral Emissions Mask measurement.

Conditions

Measurement views: Spectral Emissions Mask

Group

Read commands

Syntax

READ:SEM:SPECtrum:XY?

Arguments

None

Returns

#<num_digit><num_byte><x(1)><y(1)><x(2)><y(2)>...<x(n)><y(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<x(n)><y(n)> is the frequency (Hz) and amplitude (dBm) pair at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

READ:SEM:SPECTRUM:XY? might return #516020xxxx... (16020-byte data) for the frequency and amplitude pairs of the spectrum trace.

READ:SEM:SPECtrum[:Y]? (Query Only)

Returns the amplitude of the spectrum trace in the Spectral Emissions Mask measurement.

Conditions

Measurement views: Spectral Emissions Mask

Group

Read commands

Syntax

READ:SEM:SPECtrum[:Y]?

Arguments

None

Returns

#<num_digit><num_byte><x(1)><x(2)>...<x(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<x(n)> is the frequency (Hz) at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

READ:SEM:SPECTRUM[:Y]? might return #516020xxxx... (16020-byte data) for the frequencies of the spectrum trace.

READ:SGRam[:MEASview<y>](Query Only)

Returns the spectrogram trace data.The line is selected using the TRACe:SGRam:MEASview<y>:SELect:LINE command.

Conditions

Measurement views: Spectrogram

Group

Read commands

Syntax

READ:SGRam[:MEASview<y>]?

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Returns

#<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the trace data in dBm for the point n, 4-byte little endian floating-point format specified in IEEE 488.2.

The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples

READ:SGRam:MEASview1? might return #43204xxxx... (3204-byte data) for the spectrogram trace.

READ:SPECtrum[:MEASview<y>]:TRACe<x>? (Query Only)

Returns the trace data in the Spectrum measurement.

NOTE: TRACe5 (spectrogram) is valid when the spectrum and spectrogram measurements are running.

Conditions

Measurement views: Spectrum

Group

Read commands

Syntax

`READ:SPECtrum[:MEASview<y>]:TRACe<x>?`

Arguments

The parameter <x> = 1 to 5.

`<y>` = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the trace data in dBm for the point n, 4-byte little endian floating-point format specified in IEEE 488.2.

The unit can be changed by the `[SENSe]:POWer:UNITs` command.

Examples

`READ:SPECTRUM:MEASview1:TRACE1` might return `#43204xxxx...` (3204-byte data) for Trace 1 in the Spectrum measurement.

READ:SPECtrum[:MEASview<y>]:TRACe:BITMap? (Query Only)

Acquires a waveform and then returns trace waveform data of the bitmap trace in the Spectrum measurement.

Conditions

Measurement views: Spectrum

Group

Read commands

Syntax

`READ:SPECtrum[:MEASview<y>]:TRACe:BITMap?`

Arguments

None

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Returns

The format is a binary block of 32 bit floating point values. Each value ranges from 0 to 1.0, where 1.0 represents 100%.

Examples

READ:SPECTRUM:TRACE:BITMAP? might return #42004xxxx... (2004-byte data) for the waveform data of the bitmap trace.

READ:SPURious:CARRier:POWer? (Query Only)

Returns the carrier power in the Spurious measurement.

Conditions

Measurement views: Spurious

Group

Read commands

Syntax

READ:SPURious:CARRier:POWer?

Arguments

None

Returns

<NRf> Carrier power in dBm.

The unit can be changed by the [SENSe]:POWer:UNITs (on page 2269) command.

Examples

READ:SPURIOUS:CARRIER:POWER? might return 4.227, indicating that the carrier power is 4.227 dBm.

READ:SPURious:COUNt? (Query Only)

Returns the number of spurious signals in the Spurious measurement.

Conditions

Measurement views: Spurious

Group

Read commands

Syntax

READ:SPURious:COUNt?

Arguments

None

Returns

<NRf> The spurious count.

Examples

READ:SPURIOUS:COUNT? might return 4, indicating that the spurious count is 4.

READ:SPURious:PASS? (Query Only)

Returns the pass/fail limit test result in the Spurious measurement.

Conditions

Measurement views: Spurious

Group

Read commands

Syntax

`READ:SPURious:PASS?`

Arguments

None

Returns

0 (fail) or 1 (pass).

Examples

`READ:SPURIOUS:PASS?` might return `1`, indicating that the limit test was successful.

READ:SPURious:SPECtrum:X? (Query Only)

Returns the frequencies of the spectrum trace in the Spurious measurement.

Conditions

Measurement views: Spurious

Group

Read commands

Syntax

`READ:SPURious:SPECtrum:X?`

Arguments

None

Returns

`#<num_digit><num_byte><x(1)><x(2)>...<x(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<x(n)>` is the frequency (Hz) at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`READ:SPURIOUS:SPECTRUM:X?` might return `#516020xxxx...` (16020-byte data) for the frequencies of the spectrum trace in the Spurious measurement.

READ:SPURious:SPECtrum:XY? (Query Only)

Returns the frequency and amplitude pairs of the spectrum trace in the Spurious measurement.

Conditions

Measurement views: Spurious

Group

Read commands

Syntax

`READ:SPURious:SPECtrum:XY?`

Arguments

None

Returns

`#<num_digit><num_byte><x(1)><y(1)><x(2)><y(2)>...<x(n)><y(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<x(n)><y(n)>` is the frequency (Hz) and amplitude (dBm) pair at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

The amplitude unit can be changed by the `[SENSe]:POWer:UNITs` command.

Examples

`READ:SPURIOUS:SPECTRUM:XY?` might return `#516020xxxx...` (16020-byte data) for the frequency and amplitude pairs of the spectrum trace in the Spurious measurement.

READ:SPURious:SPECtrum[:Y]? (Query Only)

Returns the amplitudes of the spectrum trace in the Spurious measurement.

Conditions

Measurement views: Spurious

Group

Read commands

Syntax

`READ:SPURious:SPECtrum[:Y]?`

Related Commands

READ:SPURious:SPECtrum:X? (on page 1828)

Arguments

None

Returns

#<num_digit><num_byte><y(1)><y(2)>...<y(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<y(n)> is the amplitude (dBm) at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples

READ:SPURIOUS:SPECTRUM:Y might return #516020xxxx... (16020-byte data) for the amplitudes of the spectrum trace in the Spurious measurement.

READ:SPURious:SPUR<x>:AMPLitude:ABSolute? (Query Only)

Returns the absolute amplitude of the specified spurious signal in the Spurious measurement.

Conditions

Measurement views: Spurious

Group

Read commands

Syntax

READ:SPURious:SPUR<x>:AMPLitude:ABSolute?

Arguments

None

Returns

<NRf> Absolute amplitude of the specified spurious in dBm.

The unit can be changed by the [SENSe]:POWer:UNITs (on page 2269) command.

Examples

READ:SPURIOUS:SPUR1:AMPLITUDE:ABSOLUTE? might return -19.782, indicating that the absolute amplitude of Spurious #1 is -19.782 dBm.

READ:SPURious:SPUR<x>:AMPLitude:RELative? (Query Only)

Returns the relative amplitude of the specified spurious signal in the Spurious measurement.

Conditions

Measurement views: Spurious

Group

Read commands

Syntax

READ:SPURious:SPUR<x>:AMPLitude:RELative?

Arguments

None

Returns

<NRf> Relative amplitude of the specified spurious signal in dB.

Use the [SENSe]:SPURious:REFerence (on page 2410) command to set the power reference.

Examples

READ:SPURIOUS:SPUR1:AMPLITUDE:RELATIVE? might return -9.782, indicating that the relative amplitude of Spurious #1 is -9.782 dB.

READ:SPURious:SPUR<x>:FREQuency:ABSolute? (Query Only)

Returns the absolute frequency of the specified spurious signal in the Spurious measurement.

Conditions

Measurement views: Spurious

Group

Read commands

Syntax

READ:SPURious:SPUR<x>:FREQuency:ABSolute?

Arguments

None

Returns

<NRf> Absolute frequency of the spurious signal in Hz.

Examples

READ:SPURIOUS:SPUR1:FREQUENCY:ABSOLUTE? might return 2.235E+9, indicating that the absolute frequency of Spurious #1 is 2.235 GHz.

READ:SPURious:SPUR<x>:FREQuency:RELative? (Query Only)

Returns the relative frequency of the specified spurious signal to the carrier in the Spurious measurement. This command is valid when [SENSe]:SPURious:REFerence is set to CARRier.

Conditions

Measurement views: Spurious

Group

Read commands

Syntax

READ:SPURious:SPUR<x>:FREQuency:RELative?

Arguments

None

Returns

<NRf> Relative frequency of the spurious signal to the carrier in Hz.

Examples

READ:SPURIOUS:SPUR1:FREQUENCY:RELATIVE? might return 3.634E+6, indicating that the relative frequency of Spurious #1 is 3.634 MHz.

READ:SPURious:SPUR<x>:LIMit:ABSolute? (Query Only)

Returns the absolute amplitude of the limit for the specified spurious signal in the Spurious measurement.

Conditions

Measurement views: Spurious

Group

Read commands

Syntax

READ:SPURious:SPUR<x>:LIMit:ABSolute?

Arguments

None

Returns

<NRf> Absolute amplitude of the limit for the specified spurious signal in dBm.

The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples

READ:SPURIOUS:SPUR1:LIMIT:ABSOLUTE? might return -50.0, indicating that the absolute amplitude of the limit for Spurious #1 is -50 dBm.

READ:SPURious:SPUR<x>:LIMit:RELative? (Query Only)

Returns the relative amplitude of the limit for the specified spurious signal in the Spurious measurement.

Conditions

Measurement views: Spurious

Group

Read commands

Syntax

READ:SPURious:SPUR<x>:LIMit:RELative?

Arguments

None

Returns

<NRf> Relative amplitude of the limit for the specified spurious signal in dB.

Use the [SENSe]:SPURious:REFerence (on page 2410) command to set the power reference.

Examples

READ:SPURIOUS:SPUR1:LIMIT:RELATIVE? might return -10.0, indicating that the relative amplitude of the limit for Spurious #1 is -10 dB.

READ:SPURious:SPUR<x>:LIMit:VIOLation? (Query Only)

Returns whether the specified spurious signal exceeds the limit or not.

Conditions

Measurement views: Spurious

Group

Read commands

Syntax

READ:SPURious:SPUR<x>:LIMit:VIOLation?

Arguments

None

Returns

0 (under the limit) or 1 (over the limit).

Examples

READ:SPURIOUS:SPUR1:LIMIT:VIOLATION? might return 1, indicating that Spurious #1 exceeds the limit.

READ:SPURious:SPUR<x>:RANGe? (Query Only)

Returns the frequency range in which the specified spurious signal occurred.

Conditions

Measurement views: Spurious

Group

Read commands

Syntax

READ:SPURious:SPUR<x>:RANGe?

Arguments

None

Returns

<string> "A" to "T" representing Range A to T, respectively.

Examples

`READ:SPURIOUS:SPUR1:RANGE?` might return "E", indicating that Spurious #1 is in Range E.

READ:SQUality[:MEASview<y>]:FREQuency:DEViation? (Query only)

Returns the frequency deviation in the signal quality measurement.

This command is valid when `[SENSe]:DDEMod:MODulation:TYPE` is set to C4FM, FSK2, FSK4, FSK8, or FSK16.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Read commands

Syntax

`READ:SQUality[:MEASview<y>]:FREQuency:DEViation?`

Arguments

None

Returns

`<NRf>` Frequency deviation in Hz.

Examples

`READ:SQUality:MEASview1:FREQuency:DEViation?` might return 12.68E+3, indicating the frequency deviation is 12.68 kHz for measurement view 1.

READ:SQUality[:MEASview<y>]:FREQuency:DEViation:TABLe? (Query only)

Returns the number of columns and the values in the frequency deviation table for a signal quality measurement.

This command is valid when `[SENSe]:DDEMod:MODulation:TYPE` is set to C4FM, FSK2, FSK4, FSK8, or FSK16.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Read commands

Syntax

`READ:SQUality[:MEASview<y>]:FREQuency:DEViation:TABLe?`

Arguments

None

Returns

<Dev_Num>,{<Freq_dev_Max>,<Freq_dev_Min>,<Freq_dev_Avg>}

Where

`<Dev_Num>::<NR1>` is the number of columns in the frequency deviation table.

= 2 (2FSK), 4 (4FSK, C4FM), 8 (8FSK), or 16 (16FSK)

`<Freq_dev_Max>::<NRf>` is the maximum frequency deviation in Hz.

`<Freq_dev_Min>::<NRf>` is the minimum frequency deviation in Hz.

`<Freq_dev_Avg>::<NRf>` is the average frequency deviation in Hz.

The dataset `<Freq_dev_Max>,<Freq_dev_Min>,<Freq_dev_Avg>` is returned for each symbol in ascending order of its level (for example, in order of symbol -3, -1, +1, and +3 for 4FSK).

Examples

`READ:SQUality:MEASview1:FREQuency:DEViation:TABLe?` might return 2,1.257E+3,1.039E+3,1.162E+3,1.586E+3,1.298E+3,1.425E+3 for the frequency signal, populating the results table as follows.

Deviations	-1	+1
Maximum	1.257 kHz	1.586 kHz
Average	1.162 kHz	1.425 kHz
Minimum	1.039 kHz	1.298 kHz

READ:SQUality[:MEASview<y>]:FREQuency:ERRor? (Query only)

Returns the frequency error in the signal quality measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Read commands

Syntax

`READ:SQUality[:MEASview<y>]:FREQuency:ERRor?`

Arguments

None

Returns

<NRf> Frequency error in Hz.

Examples

READ:SQUality:MEASview1:FREQuency:ERRor? might return 612.043E+3, indicating that the frequency error is 612.043 kHz for measurement view 1.

READ:SQUality[:MEASview<y>]:GAIN:IMBalance? (Query only)

Returns the gain imbalance in the signal quality measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Read commands

Syntax

READ:SQUality[:MEASview<y>]:GAIN:IMBalance?

Arguments

None

Returns

<NRf> Gain imbalance in dB.

Examples

READ:SQUality:MEASview1:GAIN:IMBalance? might return -57.746E-3, indicating that the gain imbalance is -0.057746 dB for measurement view 1.

READ:SQUality[:MEASview<y>]:ORIGin:OFFSet? (Query only)

Returns the origin offset in the signal quality measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Read commands

Syntax

READ:SQUality[:MEASview<y>]:ORIGin:OFFSet?

Arguments

None

Returns

<NRf> Origin offset in dB.

Examples

READ:SQUality:MEASview1:ORIGin:OFFSet? might return -44.968, indicating that the origin offset is -44.968 dB for measurement view 1.

READ:SQUality[:MEASview<y>]:PEAK:EVM? (Query only)

Returns the peak EVM (%) in the signal quality measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Read commands

Syntax

READ:SQUality[:MEASview<y>]:PEAK:EVM?

Arguments

None

Returns

<NRf> Peak EVM in percent (%).

Examples

READ:SQUality:MEASview1:PEAK:EVM? might return 4.276, indicating that the peak EVM is 4.276% for measurement view 1.

READ:SQUality[:MEASview<y>]:PEAK:EVM:DB? (Query only)

Returns the peak EVM (dB) in the signal quality measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Read commands

Syntax

`READ:SQUality[:MEASview<y>]:PEAK:EVM:DB?`

Arguments

None

Returns

`<NRf>` Peak EVM in dB.

Examples

`READ:SQUality:MEASview1:PEAK:EVM:DB?` might return -27.358, indicating that the peak EVM is -27.358 dB for measurement view 1.

READ:SQUality[:MEASview<y>]:PEAK:EVM:DB:OFFSet? (Query only)

Returns the peak offset EVM (dB) in the signal quality measurement.

This command is valid when `[SENSe]:DDEMod:MODulation:TYPE` is set to OQPSK or SOQPSK.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Read commands

Syntax

`READ:SQUality[:MEASview<y>]:PEAK:EVM:DB:OFFSet?`

Arguments

None

Returns

`<NRf>` Peak offset EVM in dB.

Examples

`READ:SQUality:MEASview1:PEAK:EVM:DB:OFFSet?` might return `-37.624`, indicating the peak offset EVM is –37.624 dB for measurement view 1.

READ:SQUality[:MEASview<y>]:PEAK:EVM:LOCation? (Query only)

Returns the time at which the EVM is peak.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Read commands

Syntax

`READ:SQUality[:MEASview<y>]:PEAK:EVM:LOCation?`

Arguments

None

Returns

`<NRf>` The time in symbol number at which the EVM is peak.

The unit can be changed by the `[SENSe]:DDEMod:TIMe:UNITs` command.

Examples

`READ:SQUality:MEASview1:PEAK:EVM:LOCation?` might return 68.000, indicating that the EVM is peak at symbol #68.000 for measurement view 1.

READ:SQUality[:MEASview<y>]:PEAK:EVM:LOCation:OFFSet? (Query only)

Returns the time at which the offset EVM is peak.

This command is valid when `[SENSe]:DDEMod:MODulation:TYPE` is set to OQPSK or SOQPSK.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Read commands

Syntax

`READ:SQUality[:MEASview<y>]:PEAK:EVM:LOCation:OFFSet?`

Arguments

None

Returns

<NRf> The time in symbol number at which the offset EVM is peak.

The unit can be changed by the [SENSe]:DDEMod:TIMe:UNITs command.

Examples

READ:SQUality:MEASview1:PEAK:EVM:LOCation:OFFSet? might return 123.00, indicating that the offset EVM is peak at symbol #123 for measurement view 1.

READ:SQUality[:MEASview<y>]:PEAK:EVM:OFFSet? (Query only)

Returns the peak offset EVM (%) in the signal quality measurement.

This command is valid when [SENSe]:DDEMod:MODulation:TYPE is set to OQPSK or SOQPSK.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Read commands

Syntax

READ:SQUality[:MEASview<y>]:PEAK:EVM:OFFSet?

Arguments

None

Returns

<NRf> Peak offset EVM in percent (%).

Examples

READ:SQUality:MEASview1:PEAK:EVM:OFFSet? might return 1.298, indicating the peak offset EVM is 1.298% for measurement view 1.

READ:SQUality[:MEASview<y>]:PEAK:FERRor? (Query only)

Returns the peak FSK error in the signal quality measurement.

This command is valid when [SENSe]:DDEMod:MODulation:TYPE is set to FSK2, FSK4, FSK8, or FSK16.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Read commands

Syntax

`READ:SQUality[:MEASview<y>]:PEAK:FERRor?`

Arguments

None

Returns

`<NRf>` Peak FSK error in percent (%).

Examples

`READ:SQUality:MEASview1:PEAK:FERRor?` might return 9.136, indicating the peak FSK error is 9.136% for measurement view 1.

READ:SQUality[:MEASview<y>]:PEAK:MERRor? (Query only)

Returns the peak magnitude error (%) in the signal quality measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Read commands

Syntax

`READ:SQUality[:MEASview<y>]:PEAK:MERRor?`

Arguments

None

Returns

`<NRf>` Peak magnitude error in percent (%).

Examples

`READ:SQUality:MEASview1:PEAK:MERRor?` might return 3.595, indicating that the peak magnitude error is 3.595% for measurement view 1.

READ:SQUality[:MEASview<y>]:PEAK:MERRor:DB? (Query only)

Returns the peak magnitude error (dB) in the signal quality measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Read commands

Syntax

`READ:SQUality[:MEASview<y>]:PEAK:MERRor:DB?`

Arguments

None

Returns

`<NRf>` Peak magnitude error in dB.

Examples

`READ:SQUality:MEASview1:PEAK:MERRor:DB?` might return -28.583, indicating that the magnitude error is -28.583 dB for measurement view 1.

READ:SQUality[:MEASview<y>]:PEAK:MERRor:LOCation? (Query only)

Returns the time at which the magnitude error is peak.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Read commands

Syntax

`READ:SQUality[:MEASview<y>]:PEAK:MERRor:LOCation?`

Arguments

None

Returns

`<NRf>` The time in symbol number at which the magnitude error is peak.

The unit can be changed by the `[SENSe]:DDEMod:TIMe:UNITs` command.

Examples

`READ:SQUality:MEASview1:PEAK:MERRor:LOCation?` might return 68.000, indicating that the magnitude error is peak at symbol #68 for measurement view 1.

READ:SQUality[:MEASview<y>]:PEAK:PERRor? (Query only)

Returns the peak phase error in the signal quality measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Read commands

Syntax

`READ:SQUality[:MEASview<y>]:PEAK:PERRor?`

Arguments

None

Returns

`<NRf>` Peak phase error in degrees.

Examples

`READ:SQUality:MEASview1:PEAK:PERRor?` might return 1.907, indicating that the peak phase error is 1.907 ° for measurement view 1.

READ:SQUality[:MEASview<y>]:PEAK:PERRor:LOCation? (Query only)

Returns the time at which the phase error is peak.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Read commands

Syntax

`READ:SQUality[:MEASview<y>]:PEAK:PERRor:LOCation?`

Arguments

None

Returns

<NRf> The time in symbol number at which the phase error is peak.

The unit can be changed by the [SENSe]:DDEMod:TIMe:UNITs command.

Examples

READ:SQUality:MEASview1:PEAK:PERRor:LOCation? might return 68.000, indicating that the phase error is peak at symbol #68 for measurement view 1.

READ:SQUality[:MEASview<y>]:QUADrature:ERRor? (Query only)

Returns the quadrature error in the signal quality measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Read commands

Syntax

READ:SQUality[:MEASview<y>]:QUADrature:ERRor?

Arguments

None

Returns

<NRf> Quadrature error in degrees.

Examples

READ:SQUality:MEASview1:QUADrature:ERRor? might return -14.264E-3, indicating that the quadrature error is -0.014264° for measurement view 1.

READ:SQUality[:MEASview<y>]:RHO? (Query only)

Returns the ρ (waveform quality) value in the signal quality measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Read commands

Syntax

READ:SQUality[:MEASview<y>]:RHO?

Arguments

None

Returns

<NRf> ρ value.

Examples

READ:SQUality:MEASview1:RHO? might return 998.703E-3, indicating that ρ is 0.998703 for measurement view 1.

READ:SQUality[:MEASview<y>]:RMS:EVM? (Query only)

Returns the RMS EVM (%) in the signal quality measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Read commands

Syntax

READ:SQUality[:MEASview<y>]:RMS:EVM?

Arguments

None

Returns

<NRf> RMS EVM in percent (%).

Examples

READ:SQUality:MEASview1:RMS:EVM? might return 2.417, indicating that the RMS EVM is 2.417% for measurement view 1.

READ:SQUality[:MEASview<y>]:RMS:EVM:DB? (Query only)

Returns the RMS EVM (dB) in the signal quality measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Read commands

Syntax

`READ:SQUality[:MEASview<y>]:RMS:EVM:DB?`

Arguments

None

Returns

`<NRf>` RMS EVM in dB.

Examples

`READ:SQUality:MEASview1:RMS:EVM:DB?` might return -32.356, indicating that the RMS EVM is -32.356 dB for measurement view 1.

READ:SQUality[:MEASview<y>]:RMS:EVM:DB:OFFSet? (Query only)

Returns the RMS offset EVM (dB) in the signal quality measurement.

This command is valid when `[SENSe]:DDEMod:MODulation:TYPE` is set to OQPSK or SOQPSK.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Read commands

Syntax

`READ:SQUality[:MEASview<y>]:RMS:EVM:DB:OFFSet?`

Arguments

None

Returns

`<NRf>` RMS offset EVM in dB.

Examples

`READ:SQUality:MEASview1:RMS:EVM:DB:OFFSet?` might return `-41.276`, indicating the RMS offset EVM is -41.276 dB for measurement view 1.

READ:SQUality[:MEASview<y>]:RMS:EVM:OFFSet? (Query only)

Returns the RMS offset EVM (%) in the signal quality measurement.

This command is valid when `[SENSe]:DDEMod:MODulation:TYPE` is set to OQPSK or SOQPSK.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Read commands

Syntax

`READ:SQUality[:MEASview<y>]:RMS:EVM:OFFSet?`

Arguments

None

Returns

`<NRf>` RMS offset EVM in percent (%).

Examples

`READ:SQUality:MEASview1:RMS:EVM:OFFSet?` might return 0.783, indicating the RMS offset EVM is 0.783% for measurement view 1.

READ:SQUality[:MEASview<y>]:RMS:FERRor? (Query only)

Returns the RMS FSK error in the signal quality measurement.

This command is valid when `[SENSe]:DDEMod:MODulation:TYPE` is set to FSK2, FSK4, FSK8, or FSK16.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Read commands

Syntax

`READ:SQUality[:MEASview<y>]:RMS:FERRor?`

Arguments

None

Returns

`<NRf>` RMS FSK error in percent (%).

Examples

`READ:SQUality:MEASview1:RMS:FERRor?` might return 8.67, indicating the RMS FSK error is 8.67% for measurement view 1.

READ:SQUality[:MEASview<y>]:RMS:MER:DB? (Query only)

Returns the RMS MER (Modulation Error Ratio) in dB in the signal quality measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Read commands

Syntax

`READ:SQUality[:MEASview<y>]:RMS:MER:DB?`

Arguments

None

Returns

`<NRf>` RMS MER in dB.

Examples

`READ:SQUality:MEASview1:RMS:MER:DB?` might return 27.394, indicating that the RMS MER is 27.394 dB for measurement view 1.

READ:SQUality[:MEASview<y>]:RMS:MERRor? (Query only)

Returns the RMS magnitude error (%) in the signal quality measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Read commands

Syntax

`READ:SQUality[:MEASview<y>]:RMS:MERRor?`

Arguments

None

Returns

<NRf> RMS magnitude error in percent (%).

Examples

READ:SQUality:MEASview1:RMS:MERRor? might return 1.837, indicating that the RMS magnitude error is 1.837% for measurement view 1.

READ:SQUality[:MEASview<y>]:RMS:MERRor:DB? (Query only)

Returns the RMS magnitude error (dB) in the signal quality measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Read commands

Syntax

READ:SQUality[:MEASview<y>]:RMS:MERRor:DB?

Arguments

None

Returns

<NRf> RMS magnitude error in dB.

Examples

READ:SQUality:MEASview1:RMS:MERRor:DB? might return -34.706, indicating that the magnitude error is -34.706 dB for measurement view 1.

READ:SQUality[:MEASview<y>]:RMS:PERRor? (Query only)

Returns the RMS phase error in the signal quality measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Read commands

Syntax

READ:SQUality[:MEASview<y>]:RMS:PERRor?

Arguments

None

Returns

<NRf> RMS phase error in degrees.

Examples

READ:SQUality:MEASview1:RMS:PERRor? might return 893.472E-3, indicating that the RMS phase error is 0.893472° for measurement view 1.

READ:SQUality[:MEASview<y>]:SYMBol:LENGth? (Query only)

Returns the number of analyzed symbols.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Read commands

Syntax

READ:SQUality[:MEASview<y>]:SYMBol:LENGth?

Related Commands

Returns

<NR1> indicates the length of the synch word in symbols.

Examples

READ:SQUality:MEASview1:SYMBol:LENGth? might return 3, indicating the length is three symbols for measurement view 1.

READ:SQUality[:MEASview<y>]:SYMBol:RATE? (Query only)

Returns the value of the calculated symbol rate in Hz.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

It is valid when the modulation type is 2|4|8|16FSK and [:SENSe]:DDEMod:SYMBol:RATE:SEARch is ON.

Group

Read commands

Syntax

`READ:SQUality[:MEASview<y>]:SYMBol:RATE?`

Related Commands

[SENSe]:DDEMod:SYMBol:RATE:SEARch (on page 2047)

Returns

`<NRf>` is the calculated symbol rate in Hz.

Examples

`READ:SQUality:MEASview1:SYMBol:RATE?` might return 95.24E+3, indicating the calculated symbol rate is 95.24 kHz for measurement view 1.

READ:SQUality[:MEASview<y>]:SYMBol:RATE:ERRor? (Query only)

Returns the value of the symbol rate error in percent (%).

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

It is valid when the modulation type is 2|4|8|16FSK and `[:SENSe]:DDEMod:SYMBol:RATE:SEARch` is ON.

Group

Read commands

Syntax

`READ:SQUality[:MEASview<y>]:SYMBol:RATE:ERRor?`

Related Commands

[SENSe]:DDEMod:SYMBol:RATE:SEARch (on page 2047)

Returns

`<NRf>` is the symbol error percent (%).

Examples

`READ:SQUality:MEASview1:SYMBol:RATE:ERRor?` might return -0.002, indicating that the symbol rate error is -0.002% for measurement view 1.

READ:TDIagram[:MEASview<y>]:FERRor? (Query only)

Returns the frequency error in the Trellis diagram measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Trellis diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Trellis diagram).

Group

Read commands

Syntax

`READ:TDIagram[:MEASview<y>]:FERRor?`

Arguments

None

Returns

`<freq_error>::<NRf>` is the frequency error in Hz.

Examples

`READ:TDIagram:MEASview2:FERROR?` might return -10.7E+3, indicating the frequency error is -10.7 kHz.

READ:TDIagram[:MEASview<y>]:TRACe? (Query only)

Returns the Trellis diagram trace data.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Trellis diagram

Group

Read commands

Syntax

`READ:TDIagram[:MEASview<y>]:TRACe?`

Arguments

None

Returns

`#<num_digit><num_byte><Y(1)><X(1)><Y(2)><X(2)>...<Y(n)><X(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<Y(n)>` and `<X(n)>` is the phase in degrees and the time in symbols pair at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`READ:TDIagram:MEASview2:TRACe?` might return #3160xxxx... (160-byte data) for the frequency deviation versus Time trace.

READ:TOVerview[:MEASview<y>](Query Only)

Returns the trace data in the time overview.

Conditions

Measurement views: Time overview with specified Measurement view number

Group

Read commands

Syntax

`READ:TOVerview[:MEASview<y>]`

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the trace data in dBm for the point n, 4-byte little endian floating-point format specified in IEEE 488.2.

The unit can be changed by the `[SENSe]:POWer:UNITs` command.

Examples

`READ:TOVerview:MEASview1?` might return `#43204xxxx...` (3204-byte data) for the trace in the time overview.

READ:WLAN:AD:CONSte:DATA:EVM? (Query Only)

Queries the EVM value in Constellation display.

Conditions

Measurement view: 802.11ad Constellation

Group

Read commands

Syntax

`READ:WLAN:AD:CONSte:DATA:EVM?`

Arguments

None

Returns

`<NRf>` = EVM value

Examples

`READ:WLAN:AD:CONSTE:DATA:EVM?` might return `-42.482`, indicating the EVM value present in the analysis data is -42.482 dB.

READ:WLAN:AD:CONSte:DATA:EVM:PEAK? (Query Only)

Queries the EVM Peak value in Constellation display.

Conditions

Measurement view: 802.11ad Constellation

Group

Read commands

Syntax

`READ:WLAN:AD:CONSte:DATA:EVM:PEAK?`

Arguments

None

Returns

`<NRf>` = EVM peak value

Examples

`READ:WLAN:AD:CONSTE:DATA:EVM:PEAK?` might return `-35.325`, indicating the EVM Peak value is –35.325 dB.

READ:WLAN:AD:CONSte:DATA:EVM:PEAK:SYMBol? (Query Only)

Queries the EVM peak symbol value in Constellation display.

Conditions

Measurement view: 802.11ad Constellation

Group

Read commands

Syntax

`READ:WLAN:AD:CONSte:DATA:EVM:PEAK:SYMBol?`

Arguments

None

Returns

`<NRf>` = Symbol value

Examples

`READ:WLAN:AD:CONSTE:DATA:EVM:PEAK:SYMBOL?` might return `6.312`, indicating the EVM peak symbol value is 6.312.

READ:WLAN:AD:CONSte:FERRor? (Query Only)

Queries the frequency Error value in Constellation display.

Conditions

Measurement view: 802.11ad Constellation

Group

Read commands

Syntax

`READ:WLAN:AD:CONSte:FERRor?`

Arguments

None

Returns

`<NR3>` = frequency error

Examples

`READ:WLAN:AD:CONSTE:FERROR?` might return `3.344+3`, indicating the frequency error value is 3.344 kHz.

READ:WLAN:AD:CONSte:IQDC:OFFSet? (Query Only)

Queries IQ Origin Offset value in Constellation display.

Conditions

Measurement view: 802.11ad Constellation

Group

Read commands

Syntax

`READ:WLAN:AD:CONSte:IQDC:OFFSet?`

Arguments

None

Returns

`<NRf>` = IQ Origin offset value

Examples

`READ:WLAN:AD:CONSTE:IQDC:OFFSET?` might return `-51.841`, indicating the IQ Origin Offset value is –51.841 dB.

READ:WLAN:AD:EVM:TRACe? (Query Only)

Returns the EVM versus time trace data.

Conditions

Measurement views: 802.11ad EVM vs Time

Group

Read commands

Syntax

`READ:WLAN:AD:EVM:TRACe?`

Arguments

None

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the EVM value at the nth data point in percent (%), `<data(n)>` follows 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`READ:WLAN:AD:EVM:TRACE?` might return `#42036xxxx...` (2036-byte data), indicating the EVM versus time trace data.

READ:WLAN:AD:SUMMary:AVERage:BURSt:COUNt (Query Only)

Returns the current burst number out of the total burst in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

`READ:WLAN:AD:SUMMary:AVERage:BURSt:COUNt?`

Arguments

None

Returns

`<NR1>`

Examples

`READ:WLAN:AD:SUMMARY:AVERAGE:BURST:COUNT?` might return `1`, indicating the current burst count is 1 out of the total burst.

READ:WLAN:AD:SUMMary:AVERage:BURSt:TOTAl:COUNt (Query Only)

Returns the total burst count to be averaged in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

```
READ:WLAN:AD:SUMMary:AVERage:BURSt:TOTAl:COUNt?
```

Arguments

None

Returns

<NR1>

Examples

READ:WLAN:AD:SUMMARY:AVERAGE:BURST:TOTAL:COUNT? might return 1, indicating the total burst count to be averaged is 1.

READ:WLAN:AD:SUMMary:ESNR (Query Only)

Returns the value of estimated SNR value in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Analysis Summary

Group

Read commands

Syntax

```
READ:WLAN:AD:SUMMary:ESNR?
```

Arguments

None

Returns

<NRf>

Examples

READ:WLAN:AD:SUMMARY:ESNR? might return 36.856, indicating the estimated SNR value is 36.856 dB.

READ:WLAN:AD:SUMMary:EVM:DATA:AVERage (Query Only)

Returns Avg EVM (Data) value in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

`READ:WLAN:AD:SUMMary:EVM:DATA:AVERage?`

Arguments

None

Returns

`<NRf>`

Examples

`READ:WLAN:AD:SUMMARY:EVM:DATA:AVERAGE` might return `-40.7035275551`, indicating that the Avg EVM (Data) value is −40.70 dB.

READ:WLAN:AD:SUMMary:EVM:DATA:AVERage:STATus (Query Only)

Returns Avg EVM (Data) status in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

`READ:WLAN:AD:SUMMary:EVM:DATA:AVERage:STATus?`

Arguments

None

Returns

`<String>`

PASS indicates that the test passed.

FAIL indicates that the test failed.

NA indicates that the test is not applicable.

Examples

`READ:WLAN:AD:SUMMARY:EVM:DATA:AVERAGE:STATUS?` might return `PASS`, indicating that the Avg EVM (Data) test has passed.

READ:WLAN:AD:SUMMary:EVM:DATA:MAX (Query Only)

Returns Max EVM (Data) value in the 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax
`READ:WLAN:AD:SUMMary:EVM:DATA:MAX?`

Arguments
None

Returns
`<NRf>`

Examples
`READ:WLAN:AD:SUMMARY:EVM:DATA:MAX?` might return `922.1968241036E-3`, indicating the Max EVM (Data) value is 922.19 dB.

READ:WLAN:AD:SUMMary:EVM:PACKet:CEF:EVM (Query Only)
Returns the EVM value in the CEF region of the packet in the 802.11ad Summary display.

Conditions
Measurement view: 802.11ad Summary

Group
Read commands

Syntax
`READ:WLAN:AD:SUMMary:EVM:PACKet:CEF:EVM?`

Arguments
None

Returns
`<NRf>`

Examples
`READ:WLAN:AD:SUMMARY:EVM:PACKET:CEF:EVM?` might return `6.9680400193`, indicating the EVM value in the CEF region of the packet is 6.69 dB.

READ:WLAN:AD:SUMMary:EVM:PACKet:CEF:PEAK (Query Only)
Returns the peak Error occurred at the given Symbol in CEF Packet in 802.11ad Summary display.

Conditions
Measurement view: 802.11ad Summary

Group
Read commands

Syntax
`READ:WLAN:AD:SUMMary:EVM:PACKet:CEF:PEAK?`

Arguments

None

Returns

<NRf>

Examples

READ:WLAN:AD:SUMMARY:EVM:PACKET:CEF:PEAK might return 13.7763872743, indicating the peak error value at the given symbol in CEF region of the packet is 13.77 dB.

READ:WLAN:AD:SUMMary:EVM:PACKet:CEF:SYMBol (Query Only)

Returns the symbol at which the peak error occurred in the CEF region of the packet in the 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

READ:WLAN:AD:SUMMary:EVM:PACKet:CEF:SYMBol?

Arguments

None

Returns

<NRf>

Examples

READ:WLAN:AD:SUMMARY:EVM:PACKET:CEF:SYMBOL might return 7.4020000000E+3, indicating the peak error occurred at symbol 7.40 in the CEF region of the packet.

READ:WLAN:AD:SUMMary:EVM:PACKet:DATA:EVM (Query Only)

Returns the EVM value in Data region of the packet in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

READ:WLAN:AD:SUMMary:EVM:PACKet:DATA:EVM?

Arguments

None

Returns

<NRf>

Examples

`READ:WLAN:AD:SUMMARY:EVM:PACKET:DATA:EVM` might return `-33.5668742100`, indicating EVM value in Data region of the packet is -33.56 dB.

READ:WLAN:AD:SUMMary:EVM:PACKet:DATA:PEAK (Query Only)

Returns the peak Error occurred at the given Symbol in Data Packet in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

`READ:WLAN:AD:SUMMary:EVM:PACKet:DATA:PEAK?`

Arguments

None

Returns

`<NRf>`

Examples

`READ:WLAN:AD:SUMMARY:EVM:PACKET:DATA:PEAK?` might return `-26.5964736883`, indicating Peak Error value at given Symbol in Data packet is -26.59 dB.

READ:WLAN:AD:SUMMary:EVM:PACKet:DATA:STATus (Query Only)

Returns the status of the Data packet in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

`READ:WLAN:AD:SUMMary:EVM:PACKet:DATA:STATus?`

Arguments

None

Returns

`<String>`

PASS indicates that the test passed.

FAIL indicates that the test failed.

NA indicates that the test is not Applicable

Examples

`READ:WLAN:AD:SUMMARY:EVM:PACKET:DATA:STATUS?` might return `Pass`, indicating that the Data packet test result has Passed.

READ:WLAN:AD:SUMMary:EVM:PACKet:DATA:SYMBol (Query Only)

Returns the symbol at which the peak Error occurred in the DATA region of the packet in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

`READ:WLAN:AD:SUMMary:EVM:PACKet:DATA:SYMBol?`

Arguments

None

Returns

`<NRf>`

Examples

`READ:WLAN:AD:SUMMARY:EVM:PACKET:DATA:SYMBOL` might return `4.6610000000E+3`, indicating at 4.66 symbol, peak error occurred in Data Region of the packet.

READ:WLAN:AD:SUMMary:EVM:PACKet:GUARd:EVM (Query Only)

Returns the EVM value in the Guard region of the packet in the 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

`READ:WLAN:AD:SUMMary:EVM:PACKet:GUARd:EVM?`

Arguments

None

Returns

`<NRf>`

Examples

`READ:WLAN:AD:SUMMARY:EVM:PACKET:GUARD:EVM` might return `-65.4983500315`, indicating the EVM value in the Guard region of the packet is –65.49 dB.

READ:WLAN:AD:SUMMary:EVM:PACKet:GUARd:PEAK (Query Only)

Returns the peak error occurred at the given symbol in Guard packet in the 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

`READ:WLAN:AD:SUMMary:EVM:PACKet:GUARd:PEAK?`

Arguments

None

Returns

`<NRf>`

Examples

`READ:WLAN:AD:SUMMARY:EVM:PACKET:GUARD:PEAK` might return `-26.5964736883`, indicating the peak error value at the given symbol in the Guard packet is −26.59 dB.

READ:WLAN:AD:SUMMary:EVM:PACKet:GUARd:SYMBol (Query Only)

Returns the symbol at which the peak error occurred in the Guard region of the packet in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

`READ:WLAN:AD:SUMMary:EVM:PACKet:GUARd:SYMBol?`

Arguments

None

Returns

`<NRf>`

Examples

`READ:WLAN:AD:SUMMARY:EVM:PACKET:GUARD:SYMBOL` might return `3.3340000000E+3`, indicating the peak error occurred at symbol 3.33 in the Guard region of the packet.

READ:WLAN:AD:SUMMary:EVM:PACKet:HEADer:EVM (Query Only)

Returns the EVM value in the Header region of the packet in the 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

`READ:WLAN:AD:SUMMary:EVM:PACKet:HEADer:EVM?`

Arguments

None

Returns

`<NRf>`

Examples

`READ:WLAN:AD:SUMMARY:EVM:PACKET:HEADER:EVM` might return `-67.1981081169`, indicating the EVM value in the Header region of the packet is –67.19 dB.

READ:WLAN:AD:SUMMary:EVM:PACKet:HEADer:PEAK (Query Only)

Returns the peak error occurred at the given symbol in the Header packet in the 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

`READ:WLAN:AD:SUMMary:EVM:PACKet:HEADer:PEAK?`

Arguments

None

Returns

`<NRf>`

Examples

`READ:WLAN:AD:SUMMARY:EVM:PACKET:HEADER:PEAK` might return `-60.9603086224`, indicating the peak error value at the given symbol in the Header region of the packet is –60.9603086224 .

READ:WLAN:AD:SUMMary:EVM:PACKet:HEADer:SYMBol (Query Only)

Returns the symbol at which the peak error occurred in the Header region of the packet in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

`READ:WLAN:AD:SUMMary:EVM:PACKet:HEADer:SYMBol?`

Arguments

None

Returns

`<NRf>`

Examples

`READ:WLAN:AD:SUMMARY:EVM:PACKET:HEADER:SYMBOL` might return `3.3970000000E+3`, indicating the peak error occurred at symbol 3.39 in the Header region of the packet.

READ:WLAN:AD:SUMMary:EVM:PACKet:STF:EVM (Query Only)

Returns the EVM value in STF region of the packet in the 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

`READ:WLAN:AD:SUMMary:EVM:PACKet:STF:EVM?`

Arguments

None

Returns

`<NRf>`

Examples

`READ:WLAN:AD:SUMMARY:EVM:PACKET:STF:EVM` might return `-22.9729890175`, indicating the EVM value in the STF region of the packet is −22.97 dB.

READ:WLAN:AD:SUMMary:EVM:PACKet:STF:PEAK (Query Only)

Returns the peak error at the given symbol in the STF region of the packet in the 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

READ:WLAN:AD:SUMMary:EVM:PACKet:STF:PEAK?

Arguments

None

Returns

<NRf>

Examples

READ:WLAN:AD:SUMMARY:EVM:PACKET:STF:PEAK might return 38.8123661280, indicating the peak error at the given symbol in the STF packet is 38.81 dB.

READ:WLAN:AD:SUMMary:EVM:PACKet:STF:SYMBol (Query Only)

Returns the symbol at which the peak error occurred in the STF region of the packet in the 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

READ:WLAN:AD:SUMMary:EVM:PACKet:STF:SYMBol?

Arguments

None

Returns

<NRf>

Examples

READ:WLAN:AD:SUMMARY:EVM:PACKET:STF:SYMBOL might return 0.0000, indicating the peak error occurred at symbol 0.0000 in the STF region of the packet.

READ:WLAN:AD:SUMMary:FERRor:AVERage (Query Only)

Returns the Avg Frequency Error value in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

READ:WLAN:AD:SUMMary:FERRor:AVERage?

Arguments

None

Returns

<NRf>

Examples

READ:WLAN:AD:SUMMARY:FERROR:AVERAGE might return -114.1332879317E-3, indicating the Avg Frequency Error value is -114.1332879317E-3 ppm.

READ:WLAN:AD:SUMMary:FERRor:AVERage:STATus (Query Only)

Returns the Avg Frequency Error status in the 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

READ:WLAN:AD:SUMMary:FERRor:AVERage:STATus?

Arguments

None

Returns

<String>

PASS indicates that the test passed.

FAIL indicates that the test failed.

NA indicates that the test is not applicable.

Examples

READ:WLAN:AD:SUMMARY:FERROR:AVERAGE:STATUS? might return PASS, indicating that the Avg Frequency Error test has passed.

READ:WLAN:AD:SUMMary:FERRor:HERTz (Query Only)

Returns the Frequency Error value in 802.11ad Summary Display in Hz.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

READ:WLAN:AD:SUMMary:FERRor:HERTz?

Arguments

None

Returns

<NRf>

Examples

READ:WLAN:AD:SUMMARY:FERROR:HERTZ might return 2.875E+3, indicating the Frequency Error value is 2.875E+3 Hz.

READ:WLAN:AD:SUMMary:FERRor:MAX (Query Only)

Returns Max Frequency Error value in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

READ:WLAN:AD:SUMMary:FERRor:MAX?

Arguments

None

Returns

<NRf>

Examples

READ:WLAN:AD:SUMMARY:FERROR:MAX might return -114.1332879317E-3, indicating the maximum frequency error is -14.1332879317E-3 ppm.

READ:WLAN:AD:SUMMary:FERRor:PPM (Query Only)

Returns the Frequency Error value in 802.11ad Summary display in ppm.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

READ:WLAN:AD:SUMMary:FERRor:PPM?

Arguments

None

Returns

<NRf>

Examples

`READ:WLAN:AD:SUMMARY:FERROR:PPM` might return `575.023E-3`, indicating the Frequency Error value is 575.023E-3 ppm.

READ:WLAN:AD:SUMMary:FERRor:STATus (Query Only)

Returns the Frequency Error Status in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

`READ:WLAN:AD:SUMMary:FERRor:STATus?`

Arguments

None

Returns

`<String>`

PASS indicates that the test passed.

FAIL indicates that the test failed.

NA indicates that the test is not applicable.

Examples

`READ:WLAN:AD:SUMMARY:FERROR:STATUS?` might return `PASS`, indicating that the Frequency Error test has passed.

READ:WLAN:AD:SUMMary:IQ:GAIN:IMBALance (Query Only)

Returns IQ Gain Imbalance value in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

`READ:WLAN:AD:SUMMary:IQ:GAIN:IMBALance?`

Arguments

None

Returns

`<NRf>`

Examples

`READ:WLAN:AD:SUMMARY:IQ:GAIN:IMBALANCE` might return `69.2700818181E-3`, indicating IQ Gain Imbalance value is 69.2700818181E-3 % .

READ:WLAN:AD:SUMMary:IQ:ORIGin:OFFSet (Query Only)

Returns IQ Origin Offset value in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

`READ:WLAN:AD:SUMMary:IQ:ORIGin:OFFSet?`

Arguments

None

Returns

`<NRf>`

Examples

`READ:WLAN:AD:SUMMARY:IQ:ORIGIN:OFFSET` might return `-160.0000000000`, indicating that the IQ Origin Offset value is −160 dB.

READ:WLAN:AD:SUMMary:IQ:QUADrature:ERROr (Query Only)

Returns IQ Quadrature Error value in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

`READ:WLAN:AD:SUMMary:IQ:QUADrature:ERROr?`

Arguments

None

Returns

`<NRf>`

Examples

`READ:WLAN:AD:SUMMARY:IQ:QUADRATURE:ERROR?` might return `-220.5967903137E-3`, indicating IQ Quadrature Error value is −220.5967903137E-3 degrees.

READ:WLAN:AD:SUMMary:PACKet:AGGRegation (Query Only)

Returns the value of Aggregation packet information in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

```
READ:WLAN:AD:SUMMary:PACKet:AGGRegation?
```

Arguments

None

Returns

```
<Boolean> 0, 1
```

Examples

`READ:WLAN:AD:SUMMARY:PACKET:AGGREGATION?` might return `"0"`, indicating the Aggregation value is "0".

READ:WLAN:AD:SUMMary:PACKet:BTR (Query Only)

Returns the value of Beam Tracking Request packet information in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

```
READ:WLAN:AD:SUMMary:PACKet:BTR?
```

Arguments

None

Returns

```
<Boolean> 0, 1
```

Examples

`READ:WLAN:AD:SUMMARY:PACKET:BTR` might return `"0"`, indicating the Beam Tracking Request value.

READ:WLAN:AD:SUMMary:PACKet:HCS (Query Only)

Returns the value of HCS packet information in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

`READ:WLAN:AD:SUMMary:PACKet:HCS?`

Arguments

None

Returns

`<String>`

Examples

`READ:WLAN:AD:SUMMARY:PACKET:HCS` might return `"1001010000001000"`, indicating the HCS value.

READ:WLAN:AD:SUMMary:PACKet:LENGth (Query Only)

Returns the value of Length packet information in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

`READ:WLAN:AD:SUMMary:PACKet:LENGth?`

Arguments

None

Returns

`<String>`

Examples

`READ:WLAN:AD:SUMMARY:PACKET:LENGTH` might return `"1000"`, indicating the Length value is "1000".

READ:WLAN:AD:SUMMary:PACKet:MCS (Query Only)

Returns the value of MCS packet information in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

READ:WLAN:AD:SUMMary:PACKet:MCS?

Arguments

None

Returns

<NR1> Range is 0 to 12

Examples

READ:WLAN:AD:SUMMARY:PACKET:MCS might return 5, indicating the MCS value is 5.

READ:WLAN:AD:SUMMary:PACKet:PPDU (Query Only)

Returns the value of Additional PPDU packet information in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

READ:WLAN:AD:SUMMary:PACKet:PPDU?

Arguments

None

Returns

<Boolean> 0,1

Examples

READ:WLAN:AD:SUMMARY:PACKET:PPDU might return "0", indicating the Additional PPDU value is "0".

READ:WLAN:AD:SUMMary:PACKet:RSSI (Query Only)

Returns the value of last RSSI packet information in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

READ:WLAN:AD:SUMMary:PACKet:RSSI?

Arguments

None

Returns

<String>

Examples

READ:WLAN:AD:SUMMARY:PACKET:RSSI might return "0000", indicating the last RSSI value.

READ:WLAN:AD:SUMMary:PACKet:SCRAmbler:INITialization (Query Only)

Returns the value of Scrambler initialization packet information in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

READ:WLAN:AD:SUMMary:PACKet:SCRAmbler:INITialization?

Arguments

None

Returns

<String>

Examples

READ:WLAN:AD:SUMMARY:PACKET:SCRAMBLER:INITIALIZATION might return "0101110", indicating the Scrambler Initialization value is "0101110".

READ:WLAN:AD:SUMMary:PACKet:TAROUND (Query Only)

Returns the value of Turnaround packet information in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

READ:WLAN:AD:SUMMary:PACKet:TAROUND?

Arguments

None

Returns

<Boolean> 0,1

Examples

READ:WLAN:AD:SUMMARY:PACKET:TAROUND might return "1", indicating the Turnaround value.

READ:WLAN:AD:SUMMary:PACKet:TLEN (Query Only)

Returns the value of Training Length packet information in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

`READ:WLAN:AD:SUMMary:PACKet:TLEN?`

Arguments

None

Returns

`<NRf>`

Examples

`READ:WLAN:AD:SUMMARY:PACKET:TLEN` might return `21`, indicating the value of Training Length packet is 21.

READ:WLAN:AD:SUMMary:PACKet:TYPE (Query Only)

Returns the value of Packet Type information in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

`READ:WLAN:AD:SUMMary:PACKet:TYPE?`

Arguments

None

Returns

`<Boolean> 0,1`

Examples

`READ:WLAN:AD:SUMMARY:PACKET:TYPE` might return `"0"`, indicating the Packet Type value is "0".

READ:WLAN:AD:SUMMary:RCPI (Query Only)

Returns the RCPI value in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

`READ:WLAN:AD:SUMMary:RCPI?`

Arguments

None

Returns

`<NR1>`

Examples

`READ:WLAN:AD:SUMMARY:RCPI` might return `207.0000000000`, indicating the RCPI value is 207.

READ:WLAN:AD:SUMMary:RF:POWer (Query Only)

Queries the RF Output Power value in 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Read commands

Syntax

`READ:WLAN:AD:SUMMary:RF:POWer?`

Arguments

None

Returns

`<NRf>`

Examples

`READ:WLAN:AD:SUMMARY:RF:POWER` might return `-6.3434391022`, indicating the RF Output Power is -6.3434391022 dBm.

READ:WLAN:AY:CONSte:DATA:EVM? (Query Only)

Query the maximum EVM value present in the analysis data in the Constellation display.

Conditions

Measurement view: 802.11ay Analysis Constellation

Group

Read commands

Syntax
`READ:WLAN:AY:CONSte:DATA:EVM?`

Returns
`<NRf>`

Units: dB

Examples
`READ:WLAN:AY:CONS:FREQ:ERRO?` may return `-42.482`, which indicates that the maximum EVM value is -42.482 dB.

READ:WLAN:AY:CONSte:DATA:EVM:PEAK? (Query Only)
Query the Data Peak Error value in the Constellation display.

Conditions
Measurement view: 802.11ay Analysis Constellation

Group
Read commands

Syntax
`READ:WLAN:AY:CONSte:DATA:EVM:PEAK?`

Returns
`<NRf>`

Units: dB

Examples
`READ:WLAN:AY:CONS:DATA:EVM:PEAK?` may return `-35.325`, which indicates that the Data Peak Error value is -35.325 dB.

READ:WLAN:AY:CONSte:DATA:EVM:PEAK:SYMBol? (Query Only)
Query the maximum EVM value at the given Symbol in the Constellation display.

Conditions
Measurement view: 802.11ay Analysis Constellation

Group
Read commands

Syntax
`READ:WLAN:AY:CONSte:DATA:EVM:PEAK:SYMBol?`

Returns
`<NRf>`

Examples
`READ:WLAN:AY:CONS:DATA:EVM:PEAK:SYMB?` may return `3.327e+003`, which is the maximum EVM value at the 3327 Symbol.

READ:WLAN:AY:CONSte:FERRor? (Query Only)

Query the Frequency Error value in the Constellation display.

Conditions

Measurement view: 802.11ay Analysis Constellation

Group

Read commands

Syntax

`READ:WLAN:AY:CONSte:FERRor?`

Returns

`<NRf>`

Units: Hz

Examples

`READ:WLAN:AY:CONS:FERR?` may return `-51.841`, which indicates that the Frequency Error value is -51.841 Hz.

READ:WLAN:AY:CONSte:FREQuency:ERROr? (Query Only)

Query the Frequency Error value in the Constellation display.

Conditions

Measurement view: 802.11ay Analysis Constellation

Group

Read commands

Syntax

`READ:WLAN:AY:CONSte:FREQuency:ERROr?`

Returns

`<NRf>`

Units: Hz

Examples

`READ:WLAN:AY:CONS:FREQ:ERRO?` may return `0.0000`, which indicates that the Frequency Error value is 0.0000 Hz.

READ:WLAN:AY:CONSte:IQDC:OFFSet? (Query Only)

Query the IQ Origin Offset value in the Constellation display in dB.

Conditions

Measurement view: 802.11ay Analysis Constellation

Group

Read commands

Syntax

`READ:WLAN:AY:CONSte:IQDC:OFFSet?`

Returns

`<NRf>`

Examples

`READ:WLAN:AY:CONS:IQDC:OFFS?` may return `-51.841`, which indicates that the IQ Origin Offset value is -51.841 dB.

READ:WLAN:AY:EVM:TRACe? (Query Only)

Query the EVM versus time trace data after performing a single run.

NOTE: The instrument must be connected.

Conditions

Measurement views: 802.11ay Analysis EVM

Group

Read commands

Syntax

`READ:WLAN:AY:EVM:TRACe?`

Returns

`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

where:

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the EVM value at the nth data point in percentage (%).

`<data(n)>` follows 4-byte little endian floating point format specified in IEEE 488.2.

Examples

`READ:WLAN:AY:EVM:TRAC?` may return #42036xxxx...(2036-byte data) for the EVM versus time trace.

READ:WLAN:AY:SUMMary:AVERage:BURSt:COUNt? (Query Only)

Query the current Burst Count in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

`READ:WLAN:AY:SUMMary:AVERage:BURSt:COUNt?`

Returns

<NR1>

Examples

READ:WLAN:AY:SUMM:AVER:BURS:COUN? may return 1, which indicates that the current Burst Count is 1.

READ:WLAN:AY:SUMMary:AVERage:BURSt:TOTAl:COUNt? (Query Only)

Query the Total Burst Count to be averaged.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

READ:WLAN:AY:SUMMary:AVERage:BURSt:TOTAl:COUNt?

Returns

<NR1>

Examples

READ:WLAN:AY:SUMM:AVER:BURS:TOTA:COUN? may return 1, which indicates that the Total Burst Count to be averaged is 1.

READ:WLAN:AY:SUMMary:ESNR? (Query Only)

Query the Estimated SNR (Signal-to-Noise Ratio) value in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

READ:WLAN:AY:SUMMary:ESNR?

Returns

<NRf>

Units: dB

Examples

READ:WLAN:AY:SUMM:ESNR? may return 34.50, which indicates that the Estimated SNR value is 34.50 dB.

READ:WLAN:AY:SUMMary:EVM:DATA:AVERage? (Query Only)

Query the Average EVM (Data) value in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

`READ:WLAN:AY:SUMMary:EVM:DATA:AVERage?`

Returns

`<NRf>`

Units: dB or % (based on the selected EVM unit)

Examples

`READ:WLAN:AY:SUMM:EVM:DATA:AVER?` may return `-40.7035275551`, which indicates that the Average EVM (Data) value is -40.7035275551 dB.

READ:WLAN:AY:SUMMary:EVM:DATA:AVERage:STATus? (Query Only)

Query the Average EVM (Data) Status in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

`READ:WLAN:AY:SUMMary:EVM:DATA:AVERage:STATus?`

Returns

`<string>`

Examples

`READ:WLAN:AY:SUMM:EVM:DATA:AVER:STAT?` may return `PASS`, `FAIL`, or `NA`.

READ:WLAN:AY:SUMMary:EVM:DATA:MAX? (Query Only)

Query the Maximum EVM (Data) value in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

`READ:WLAN:AY:SUMMary:EVM:DATA:MAX?`

Returns

<NRf>

Units: dB or % (based on the selected EVM unit)

Examples

READ:WLAN:AY:SUMM:EVM:DATA:MAX? may return 922.1968241036e-3, which indicates that the Maximum EVM (Data) value is 922.1968241036e-3 dB.

READ:WLAN:AY:SUMMary:EVM:DATA:SIGma? (Query Only)

Query the Standard Deviation (1-Sigma) of the EVM Data.

When the EVM unit is in dB, it queries the Standard Deviation after converting evm values to the dB scale.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

READ:WLAN:AY:SUMMary:EVM:DATA:SIGma?

Returns

<NRf>

Units: dB

Examples

READ:WLAN:AY:SUMM:EVM:DATA:SIG? may return 1.8950, which indicates that the 1-Sigma EVM Data value is 1.8950 dB.

READ:WLAN:AY:SUMMary:EVM:PACKet:DATA:EVM? (Query Only)

Query the EVM value that occurs in the Data Region of the packet in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

READ:WLAN:AY:SUMMary:EVM:PACKet:DATA:EVM?

Returns

<NRf>

Units: dB or % (based on the EVM unit selected)

Examples

READ:WLAN:AY:SUMM:EVM:PACK:DATA:EVM? may return -33.5668742100, which indicates that the EVM value is -33.5668742100%.

READ:WLAN:AY:SUMMary:EVM:PACKet:DATA:PEAK? (Query Only)

Query the value of the Peak Error that occurs at the given Symbol in the Data Region of the packet in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

READ:WLAN:AY:SUMMary:EVM:PACKet:DATA:PEAK?

Returns

<NRf>

Units: dB or % (based on the EVM unit selected)

Examples

READ:WLAN:AY:SUMM:EVM:PACK:DATA:PEAK? may return -26.5964736883, which indicates that the value of the Peak Error is -26.5964736883 at the given Symbol.

READ:WLAN:AY:SUMMary:EVM:PACKet:DATA:STATus? (Query Only)

Query the status of the Data Region of the packet in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

READ:WLAN:AY:SUMMary:EVM:PACKet:DATA:STATus?

Returns

<string>

Examples

READ:WLAN:AY:SUMM:EVM:PACK:DATA:STAT? may return PASS, FAIL, or NA.

READ:WLAN:AY:SUMMary:EVM:PACKet:DATA:SYMBol? (Query Only)

Query the Symbol at which the Peak Error occurs in the Data Region of the packet in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

```
READ:WLAN:AY:SUMMary:EVM:PACKet:DATA:SYMBol?
```

Returns

<NRf>

Examples

READ:WLAN:AY:SUMM:EVM:PACK:DATA:SYMB? may return 4.6610000000e+3, which indicates that the Peak Error occurs at the 4.6610000000e+3 Symbol.

READ:WLAN:AY:SUMMary:EVM:PACKet:EDMG:CEF:EVM? (Query Only)

Query the EVM value in the 11ay 2.16 GHz BW and EDMG STF for 11ay 4.32 GHz BW of the packet in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

```
READ:WLAN:AY:SUMMary:EVM:PACKet:EDMG:CEF:EVM?
```

Returns

<NRf>

Units: dB or % (based on the EVM unit selected)

Examples

READ:WLAN:AY:SUMM:EVM:PACK:EDMG:CEF:EVM? may return 6.9680400193, which indicates that the EVM value is -6.9680400193 dB.

READ:WLAN:AY:SUMMary:EVM:PACKet:EDMG:CEF:PEAK? (Query Only)

Query the Peak Error that occurs at the given Symbol in the 11ay 2.16 GHz BW and EDMG CEF for 11ay 4.32 GHz BW of the packet in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

```
READ:WLAN:AY:SUMMary:EVM:PACKet:EDMG:CEF:PEAK?
```

Returns

<NRf>

Units: dB or % (based on the EVM unit selected)

Examples

`READ:WLAN:AY:SUMM:EVM:PACK:EDMG:CEF:PEAK?` may return `13.7763872743`, which indicates that the Peak Error value is 13.7763872743% at the given Symbol.

READ:WLAN:AY:SUMMary:EVM:PACKet:EDMG:CEF:SYMBol? (Query Only)

Query the Symbol at which the Peak Error occurs in the 11ay 2.16 GHz BW and EDMG CEF for 11ay 4.32 GHz BW of the packet in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

```
READ:WLAN:AY:SUMMary:EVM:PACKet:EDMG:CEF:SYMBol?
```

Returns

<NRf>

Examples

`READ:WLAN:AY:SUMM:EVM:PACK:EDMG:CEF:SYMB?` may return `7.4020000000e+3`, which indicates that the Peak Error occurs at the 7.4020000000e+3 Symbol.

READ:WLAN:AY:SUMMary:EVM:PACKet:EDMG:STF:EVM? (Query Only)

Query the EVM value in the 11ay 2.16 GHz BW and EDMG STF for 11ay 4.32 GHz BW of the packet in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

```
READ:WLAN:AY:SUMMary:EVM:PACKet:EDMG:STF:EVM?
```

Returns

<NRf>

Units: dB or % (based on the EVM unit selected)

Examples

READ:WLAN:AY:SUMM:EVM:PACK:EDMG:STF:EVM? may return -22.9729890175, which indicates that the EVM value is -22.9729890175 dB.

READ:WLAN:AY:SUMMary:EVM:PACKet:EDMG:STF:PEAK? (Query Only)

Query the Peak Error that occurs at the given Symbol in the 11ay 2.16 GHz BW and EDMG STF for 11ay 4.32 GHz BW of the packet in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

READ:WLAN:AY:SUMMary:EVM:PACKet:EDMG:STF:PEAK?

Returns

<NRf>

Units: dB or % (based on the EVM unit selected)

Examples

READ:WLAN:AY:SUMM:EVM:PACK:EDMG:STF:PEAK? may return 38.8123661280, which indicates that the Peak Error value is 38.8123661280% at the given Symbol.

READ:WLAN:AY:SUMMary:EVM:PACKet:EDMG:STF:SYMBol? (Query Only)

Query the Symbol at which the Peak Error occurs in the 11ay 2.16 GHz BW and EDMG STF for 11ay 4.32 GHz BW of the packet in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

READ:WLAN:AY:SUMMary:EVM:PACKet:EDMG:STF:SYMBol?

Returns

<NRf>

Examples

`READ:WLAN:AY:SUMM:EVM:PACK:EDMG:STF:SYMB?` may return `0.0000`, which indicates that the Peak Error occurs at the 0.0000 Symbol.

READ:WLAN:AY:SUMMary:EVM:PACKet:GUARd:EVM? (Query Only)

Query the EVM value in the Guard Region of the packet in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

`READ:WLAN:AY:SUMMary:EVM:PACKet:GUARd:EVM?`

Returns

`<NRf>`

Units: dB or % (based on the EVM unit selected)

Examples

`READ:WLAN:AY:SUMM:EVM:PACK:GUAR:EVM?` may return `-65.4983500315`, which indicates that the value of the EVM is -65.4983500315 dB.

READ:WLAN:AY:SUMMary:EVM:PACKet:GUARd:PEAK? (Query Only)

Query the value of the Peak Error that occurs at the given Symbol in the Guard Region of the packet in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

`READ:WLAN:AY:SUMMary:EVM:PACKet:GUARd:PEAK?`

Returns

`<NRf>`

Units: dB or % (based on the EVM unit selected)

Examples

`READ:WLAN:AY:SUMM:EVM:PACK:GUAR:PEAK?` may return `-26.5964736883`, which indicates that the value of the Peak Error is -26.5964736883 at the given Symbol.

READ:WLAN:AY:SUMMary:EVM:PACKet:GUARd:SYMBol? (Query Only)

Query the Symbol at which the Peak Error occurs in the Guard Region of the packet in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

`READ:WLAN:AY:SUMMary:EVM:PACKet:GUARd:SYMBol?`

Returns

`<NRf>`

Units: dB or % (based on the EVM unit selected)

Examples

`READ:WLAN:AY:SUMM:EVM:PACK:GUAR:SYMB?` may return `3.3340000000e+3`, which indicates that the Peak Error occurs at the 3.3340000000e+3 Symbol.

READ:WLAN:AY:SUMMary:EVM:PACKet:HEADer:EVM? (Query Only)

Query the EVM value in the Header Region of the packet (excluding GI symbols) in the Summary display. This command is not applicable for 11ay 4.32 GHz BW and the value can be ignored.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

`READ:WLAN:AY:SUMMary:EVM:PACKet:HEADer:EVM?`

Returns

`<NRf>`

Units: dB or % (based on the EVM unit selected)

Examples

`READ:WLAN:AY:SUMM:EVM:PACK:HEAD:EVM?` may return `-67.1981081169`, which indicates that the EVM value in the Header Region is -67.1981081169 dB.

READ:WLAN:AY:SUMMary:EVM:PACKet:HEADer:PEAK? (Query Only)

Query the Peak Error that occurs at the given Symbol in Header Region (excluding GI symbols) of the packet in the Summary display. This command is not applicable for 11ay 4.32 GHz BW and the value can be ignored.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

`READ:WLAN:AY:SUMMary:EVM:PACKet:HEADer:PEAK?`

Returns

`<NRf>`

Units: dB or % (based on the EVM unit selected)

Examples

`READ:WLAN:AY:SUMM:EVM:PACK:HEAD:PEAK?` may return `-60.9603086224`, which indicates that the Peak Error value is -60.9603086224 dB at the given Symbol.

READ:WLAN:AY:SUMMary:EVM:PACKet:HEADer:SYMBol? (Query Only)

Query the Symbol at which the Peak Error occurs in the Header Region of the packet (excluding GI symbols) in the Summary display. This command is not applicable for 11ay 4.32 GHz BW and the value can be ignored.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

`READ:WLAN:AY:SUMMary:EVM:PACKet:HEADer:SYMBol?`

Returns

`<NRf>`

Units: dB or % (based on the EVM unit selected)

Examples

`READ:WLAN:AY:SUMM:EVM:PACK:HEAD:SYMB?` may return `3.3970000000e+3`, which indicates that the peak error occurs at the 3.3970000000e+3 Symbol.

READ:WLAN:AY:SUMMary:EVM:PACKet:PRE1Edmg:A1Header:EVM? (Query Only)

Query the value of the Pre-EDGM1 Header-A EVMpacket information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

```
READ:WLAN:AY:SUMMary:EVM:PACKet:PRE1Edmg:A1Header:EVM?
```

Returns

```
<string>
```

Units: dB or % (based on the EVM unit selected)

Examples

`READ:WLAN:AY:SUMM:EVM:PACK:PRE1E:A1H:EVM?` may return `-34.3702`, which indicates that the value of the Pre-EDGM1 Header-A EVM packet is -34.3702 dB.

READ:WLAN:AY:SUMMary:EVM:PACKet:PRE1Edmg:CEF1:EVM? (Query Only)

Query the value of the Pre-EDMG1 CEF EVM packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

```
READ:WLAN:AY:SUMMary:EVM:PACKet:PRE1Edmg:CEF1:EVM?
```

Returns

```
<string>
```

Units: dB or % (based on the EVM unit selected)

Examples

`READ:WLAN:AY:SUMM:EVM:PACK:PRE1E:CEF1:EVM?` may return `-34.5666`, which indicates that the value of the Pre-EDMG1 CEF EVM packet is -34.5666 dB.

READ:WLAN:AY:SUMMary:EVM:PACKet:PRE1Edmg:GUARd1:EVM? (Query Only)

Query the value of the Pre-EDMG1 EVM Guard packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

```
READ:WLAN:AY:SUMMary:EVM:PACKet:PRE1Edmg:GUARd1:EVM?
```

Returns

<string>

Units: dB or % (based on the EVM unit selected)

Examples

READ:WLAN:AY:SUMM:EVM:PACK:PRE1ELGUAR1:EVM? may return –34.1327, which indicates that the value of the Pre-EDMG1 EVM Guard Region is -34.1327 dB.

READ:WLAN:AY:SUMMary:EVM:PACKet:PRE1Edmg:HEADer1:EVM? (Query Only)

Query the value of the Pre-EDMG1 L-Header EVM packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

READ:WLAN:AY:SUMMary:EVM:PACKet:PRE1Edmg:HEADer1:EVM?

Returns

<string>

Units: dB or % (based on the EVM unit selected)

Examples

READ:WLAN:AY:SUMM:EVMPACK:PRE1E:HEAD1:EVM? may return –34.4326, which indicates that the value of the Pre-EDMG1 L-Header EVM packet is -34.4326 dB.

READ:WLAN:AY:SUMMary:EVM:PACKet:PRE1Edmg:STF1:EVM? (Query Only)

Query the value of the Pre-EDMG1 STF EVM packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

READ:WLAN:AY:SUMMary:EVM:PACKet:PRE1Edmg:STF1:EVM?

Returns

<string>

Units: dB or % (based on the EVM unit selected)

Examples

`READ:WLAN:AY:SUMM:EVM:PACK:PRE1E:STF1:EVM?` may return `-33.6325`, which indicates that the value of the Pre-EDMG1 STF1 EVM packet is -33.6325 dB.

READ:WLAN:AY:SUMMary:EVM:PACKet:PRE2Edmg:A2Header:EVM? (Query Only)

Query the value of the Pre-EDMG2 EDMG Header-A EVM packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

`READ:WLAN:AY:SUMMary:EVM:PACKet:PRE2Edmg:A2Header:EVM?`

Returns

`<string>`

Examples

`READ:WLAN:AY:SUMM:EVM:PACK:PRE2E:A2H:EVM?` may return `-34.37`, which indicates that the value of the Pre-EDMG2 EDMG-Header-A EVM packet is -34.37 dB.

READ:WLAN:AY:SUMMary:EVM:PACKet:PRE2Edmg:CEF2:EVM? (Query Only)

Query the value of the Pre-EDMG2 CEF EVM packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

`READ:WLAN:AY:SUMMary:EVM:PACKet:PRE2Edmg:CEF2:EVM?`

Returns

`<string>`

Units: dB or % (based on the EVM unit selected)

Examples

`READ:WLAN:AY:SUMM:EVM:PACK:PRE2E:CEF2:EVM?` may return `-34.5665`, which indicates that the value of the Pre-EDMG2 CEF EVM packet is -34.5665 dB.

READ:WLAN:AY:SUMMary:EVM:PACKet:PRE2Edmg:GUARd2:EVM? (Query Only)

Query the value of the Pre-EDMG2 Guard EVM packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

`READ:WLAN:AY:SUMMary:EVM:PACKet:PRE2Edmg:GUARd2:EVM?`

Returns

`<string>`

Examples

`READ:WLAN:AY:SUMM:EVM:PACK:PRE2E:GUAR:EVM?` may return `-33.137`, which indicates that the value of the Pre-EDMG2 Guard EVM packet is -33.137 dB.

READ:WLAN:AY:SUMMary:EVM:PACKet:PRE2Edmg:HEADer2:EVM? (Query Only)

Query the value of the Pre-EDMG2 L-Header EVM packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

`READ:WLAN:AY:SUMMary:EVM:PACKet:PRE2Edmg:HEADer2:EVM?`

Returns

`<string>`

Units: dB or % (based on the EVM unit selected)

Examples

`READ:WLAN:AY:SUMM:EVM:PACK:PRE2E:HEAD2:EVM?` may return `-34.4321`, which indicates that the value of the Pre-EDMG2 L-Header EVM packet is -34.4321 dB.

READ:WLAN:AY:SUMMary:EVM:PACKet:PRE2Edmg:STF2:EVM? (Query Only)

Query the value of the Pre-EDMG2 STF EVM packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax
READ:WLAN:AY:SUMMary:EVM:PACKet:PRE2Edmg:STF2:EVM?

Returns
<string>

Units: dB or % (based on the EVM unit selected)

Examples
READ:WLAN:AY:SUMM:EVM:PACK:PRE2E:STF2:EVM? may return −33.6324, which indicates that the value of the Pre-EDMG2 STF2 EVM packet is -33.6324 dB.

READ:WLAN:AY:SUMMary:FERRor:AVERage? (Query Only)
Query the Average Frequency Error in the Summary display.

Conditions
Measurement view: 802.11ay Analysis Summary

Group
Read commands

Syntax
READ:WLAN:AY:SUMMary:FERRor:AVERage?

Returns
<NRf>

Units: ppm

Examples
READ:WLAN:AY:SUMM:FERR:AVER? may return −114.1332879317e−3, which indicates that the Average Frequency Error value is -114.1332879317e-3 ppm.

READ:WLAN:AY:SUMMary:FERRor:AVERage:STATus? (Query Only)
Query the Average Frequency Error status in the Summary display.

Conditions
Measurement view: 802.11ay Analysis Summary

Group
Read commands

Syntax
READ:WLAN:AY:SUMMary:FERRor:AVERage:STATus?

Returns
<string>

Examples
READ:WLAN:AY:SUMM:FERR:AVER:STAT? may return PASS, FAIL, or NA.

READ:WLAN:AY:SUMMary:FERRor:HERTz? (Query Only)

Query the Frequency Error value in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

`READ:WLAN:AY:SUMMary:FERRor:HERTz?`

Returns

`<NRf>`

Units: Hz or kHz

Examples

`READ:WLAN:AY:SUMM:FERR:HERT?` may return `2.875e+3`, which indicates that the Frequency Error value is 2.875 kHz.

READ:WLAN:AY:SUMMary:FERRor:MAX? (Query Only)

Query the Maximum Frequency Error value in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

`READ:WLAN:AY:SUMMary:FERRor:MAX?`

Returns

`<NRf>`

Units: ppm

Examples

`READ:WLAN:AY:SUMM:FERR:MAX?` may return `-114.1332879317e-3`, which indicates that the Maximum Frequency Error value is -114.1332879317e-3 ppm.

READ:WLAN:AY:SUMMary:FERRor:PPM? (Query Only)

Query the Frequency Error value in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

```
READ:WLAN:AY:SUMMary:FERRor:PPM?
```

Returns

<NRf>

Units: ppm

Examples

READ:WLAN:AY:SUMM:FERR:PPM? may return 575.023e-3, which indicates that the Frequency Error value is 575.023e-3 ppm.

READ:WLAN:AY:SUMMary:FERRor:STATus? (Query Only)

Query the Frequency Error status in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

```
READ:WLAN:AY:SUMMary:FERRor:STATus?
```

Returns

<string>

Examples

READ:WLAN:AY:SUMM:FERR:STAT? may return PASS, FAIL, or NA.

READ:WLAN:AY:SUMMary:IQ:GAIN:IMBALance? (Query Only)

Query the IQ Gain Imbalance in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

```
READ:WLAN:AY:SUMMary:IQ:GAIN:IMBALance?
```

Returns

<NRf>

Units: dB

Examples

`READ:WLAN:AY:SUMM:IQ:GAIN:IMBAL?` may return `69.2700818181e-3`, which indicates that the IQ Gain Imbalance value is 69.2700818181e-3 dB.

READ:WLAN:AY:SUMMary:IQ:ORIGin:OFFSet? (Query Only)

Query the IQ Origin Offset value in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

`READ:WLAN:AY:SUMMary:IQ:ORIGin:OFFSet?`

Returns

`<NRf>`

Units: dB

Examples

`READ:WLAN:AY:SUMM:IQ:ORIG:OFFS?` may return `-160.0000000000`, which indicates that the IQ Origin Offset value is -160.0000000000 dB.

READ:WLAN:AY:SUMMary:IQ:QUADrature:ERROr? (Query Only)

Query the IQ Quadrature Error value (degrees) in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

`READ:WLAN:AY:SUMMary:IQ:QUADrature:ERROr?`

Returns

`<NRf>`

Units: degrees

Examples

`READ:WLAN:AY:SUMM:IQ:QUAD:ERRO?` may return `-220.5967903137e-3`, which indicates that the IQ Quadrature Error value is -220.5967903137e-3º degrees.

READ:WLAN:AY:SUMMary:PACKet:BEAMformed? (Query Only)

Query the value of the Beamformed packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

`READ:WLAN:AY:SUMMary:PACKet:BEAMformed?`

Returns

`<string>`

Examples

`READ:WLAN:AY:SUMM:PACK:BEAM?` may return `0`, which indicates that the value of the Beamformed packet is 0.

READ:WLAN:AY:SUMMary:PACKet:BWBitmap? (Query Only)

Query the value of the Bandwidth BitMap packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

`READ:WLAN:AY:SUMMary:PACKet:BWBitmap?`

Returns

`<string>`

Examples

`READ:WLAN:AY:SUMM:PACK:BWB?` may return `11000000`, which indicates that the value of the Bandwidth BitMap packet is 11000000.

READ:WLAN:AY:SUMMary:PACKet:COMPbw? (Query Only)

Query the value of the Compressed BW packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

`READ:WLAN:AY:SUMMary:PACKet:COMPbw?`

Returns

`<string>`

Examples

`READ:WLAN:AY:SUMM:PACK:COMP?` may return `2`, which indicates that the value of the Compressed BW packet is 2.

READ:WLAN:AY:SUMMary:PACKet:CRC? (Query Only)

Query the value of the CRC packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

`READ:WLAN:AY:SUMMary:PACKet:CRC?`

Returns

`<string>`

Examples

`READ:WLAN:AY:SUMM:PACK:CRC?` may return `0110001001011010`, which indicates that the value of the CRC packet is 0110001001011010.

READ:WLAN:AY:SUMMary:PACKet:EDMGmcs? (Query Only)

Query the value of the EDMG MCS packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

`READ:WLAN:AY:SUMMary:PACKet:EDMGmcs?`

Returns

`<NRf>`

Examples

`READ:WLAN:AY:SUMM:PACK:EDMG?` may return `5`, which indicates that the value of the EDMG MCS packet is 5.

READ:WLAN:AY:SUMMary:PACKet:LDPCcwlen? (Query Only)

Query the value of the additional LPDC CW Length packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

`READ:WLAN:AY:SUMMary:PACKet:LDPCcwlen?`

Returns

`<string>`

Examples

`READ:WLAN:AY:SUMM:PACK:LDPC?` may return `0`, which indicates that the value of the LDPC CW Length packet is 0.

READ:WLAN:AY:SUMMary:PACKet:PRIMarychannel? (Query Only)

Query the value of the Primary Channel packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

`READ:WLAN:AY:SUMMary:PACKet:PRIMarychannel?`

Returns

`<string>`

Examples

`READ:WLAN:AY:SUMM:PACK:PRIM?` may return `0`, which indicates that the value of the Primary Channel packet is 0.

READ:WLAN:AY:SUMMary:PACKet:PSDUlen? (Query Only)

Query the Length value of the packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

`READ:WLAN:AY:SUMMary:PACKet:PSDUlen?`

Returns

`<string>`

Examples

`READ:WLAN:AY:SUMM:PACK:PSDU?` may return `1000`, which indicates that the Length value of the packet is 1000.

READ:WLAN:AY:SUMMary:PACKet:RSSI? (Query Only)

Query the value of the last RSSI packet information in the Summary display.

The first 2 bits of the last RSSI will fetch the Guard Interval type for EDMG (00/01/10 - Short/Normal/Long).

Conditions

Measurement view 802.11ay Analysis Summary

Group

Read commands

Syntax

`READ:WLAN:AY:SUMMary:PACKet:RSSI?`

Returns

`<string>`

Examples

`READ:WLAN:AY:SUMM:PACK:RSSI?` may return `0000`, which indicates that the last value of the RSSI packet is 0000.

READ:WLAN:AY:SUMMary:PACKet:SCRAmbler:INITialization? (Query Only)

Query the value of the Scrambler Initialization packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

`READ:WLAN:AY:SUMMary:PACKet:SCRAmbler:INITialization?`

Returns

`<string>`

Examples

`READ:WLAN:AY:SUMM:PACK:SCRA:INIT?` may return `0101110`, which indicates that the value of the Scrambler Initialization packet is 0101110.

READ:WLAN:AY:SUMMary:PACKet:SPATialstreams? (Query Only)

Query the value of the Spatial Streams packet information in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax
READ:WLAN:AY:SUMMary:PACKet:SPATialstreams?

Returns
<string>

Examples
READ:WLAN:AY:SUMM:PACK:SPAT? may return 0, which indicates that the value of the Spatial Streams packet is 0.

READ:WLAN:AY:SUMMary:PACKet:STBCapplied? (Query Only)
Query the value of the STBC Applied packet information in the Summary display.

Conditions
Measurement view: 802.11ay Analysis Summary

Group
Read commands

Syntax
READ:WLAN:AY:SUMMary:PACKet:STBCapplied?

Returns
<string>

Examples
READ:WLAN:AY:SUMM:PACK:STBC? may return 0, which indicates that the value of the STBC Applied packet is 0.

READ:WLAN:AY:SUMMary:PACKet:TRNLen? (Query Only)
Query the value of the EDMG TRN Length packet information in the Summary display.

Conditions
Measurement view: 802.11ay Analysis Summary

Group
Read commands

Syntax
READ:WLAN:AY:SUMMary:PACKet:TRNLen?

Returns
<string>

Examples
READ:WLAN:AY:SUMM:PAC:TRNL? may return 0, which indicates that the value of the EDMG TRN Length packet is 0.

READ:WLAN:AY:SUMMary:RCPI? (Query Only)
Query the RCPI (Received Channel Power Indicator) value in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

`READ:WLAN:AY:SUMMary:RCPI?`

Returns

`<NR1>`

Examples

`READ:WLAN:AY:SUMM:RCPI?` may return `207.0000000000`, which indicates that the RCPI value is 207.0000000000.

READ:WLAN:AY:SUMMary:RF:POWer? (Query Only)

Query the Radio Frequency (RF) Output Power value in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

`READ:WLAN:AY:SUMMary:RF:POWer?`

Returns

`<NRf>`

You must select the units in the Units tab.

Examples

`READ:WLAN:AY:SUMM:RF:POW?` may return `-6.3434391022`, which indicates that the Radio Frequency Output Power value is -6.3434391022.

READ:WLAN:AY:SUMMary:SIG:QUAlity? (Query Only)

Query the Signal Quality value in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Read commands

Syntax

`READ:WLAN:AY:SUMMary:SIG:QUAlity?`

Returns

<NRf>

Units: rho

Examples

READ:WLAN:AY:SUMM:SIG:QUA? may return 999.6323e-3,, which indicates that the Signal Quality value is 999.6323e-3 rho.

READ:WLAN:CONSte:MAGNitude? (Query Only)

Returns the magnitude data for the WLAN constellation measurement.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

READ:WLAN:CONSte:MAGNitude?

Returns

Returned data is in the form

#<num_digit><num_byte><data(1)><data(2)>..<data(n)>

where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point in dB, 4-byte little endian floating-point format specified in IEEE 488.2

Examples

READ:WLAN:CONS:MAGN? might return #510400.. (10400-byte data) for the measurement.

READ:WLAN:CONSte:PHASe? (Query Only)

Returns the phase data for the WLAN Constellation measurement.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

READ:WLAN:CONSte:PHASe?

Returns

Returned data is in the form

#<num_digit><num_byte><data(1)><data(2)>..<data(n)>

where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point in dB, 4-byte little endian floating-point format specified in IEEE 488.2

Examples

READ:WLAN:CONS:PHAS? might return #510400.. (10400-byte data) for the measurement.

READ:WLAN:CONSte:TYPE? (Query Only)

Returns the type (context) of the WLAN Constellation measurement.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

READ:WLAN:CONSte:TYPE?

Returns

#<num_digit><num_byte><data(1)><data(2)>..<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point in 16-bit little endian integers.

The integers must be translated to the proper context as listed below.

Integer value Type (context):

- 0 Pilot
- 1 Data
- 2 Unused or Null

Examples

READ:WLAN:CONS:TYPE? might return #41352.. (1352-byte data) for the measurement. The actual data must be decoded to the context type using the table above.

READ:WLAN:CONSte:VALue? (Query Only)

Returns the value (decoded symbol) for the WLAN Constellation measurement.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

`READ:WLAN:CONSte:VALue?`

Returns

`#<num_digit><num_byte><data(1)><data(2)>..<data(n)>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the nth data point in 16-bit little endian integers.

Examples

`READ:WLAN:CONSTE:VALUE?` might return #41352 indicating 1352 bytes of data for the measurement.

READ:WLAN:CRESponse:MAGNitude? (Query Only)

Returns the magnitude data for the WLAN Channel Response measurement.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

`READ:WLAN:CRESponse:MAGNitude?`

Returns

Returned data is in the form

`#<num_digit><num_byte><data(1)><data(2)>..<data(n)>`

where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the nth data point in dB, 4-byte little endian floating-point format specified in IEEE 488.2

Examples

`READ:WLAN:CRES:MAGN?` might return #510400.. (10400-byte data) for the measurement.

READ:WLAN:CRESponse:PHASe? (Query Only)

Returns the phase data for the WLAN Channel Response measurement.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

`READ:WLAN:CRESponse:PHASe?`

Returns

Returned data is in the form

`#<num_digit><num_byte><data(1)><data(2)>..<data(n)>`

where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the nth data point in dB, 4-byte little endian floating-point format specified in IEEE 488.2

Examples

`READ:WLAN:CRES:PHAS?` might return `#510400` (10400-byte data) representing the phase data for the measurement.

READ:WLAN:EVM:TRACe<x>? (Query Only)

Returns the trace data for the selected WLAN EVM trace.

When <x> is 1, the parameter is Matrix.

When <x> is 2, the parameter is Average versus Symbol.

When <x> is 3, the parameter is Average versus Subcarrier

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

`READ:WLAN:EVM:TRACe<x>?`

Returns

`#<num_digit><num_byte><data(1)><data(2)>..<data(n)>`

Where:

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`READ:WLAN:EVM:TRAC2?` might return `#510400..` (10400-byte data) for the measurement.

READ:WLAN:FLATness:PASS? (Query Only)

Queries whether the average power level values of the subcarriers across the signal bandwidth on the WLAN Spectral Flatness display remain within the limits defined for a particular standard.

NOTE: WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.

Group

Read commands

Syntax

`READ:WLAN:FLATness:PASS?`

Returns

Returns "1" in the case that all data remains within the defined limits, and "0" otherwise.

Examples

`READ:WLAN:FLAT:PASS?` might return `1`, which means that the average power level values of subcarriers across the signal bandwidth remains within the defined limits.

READ:WLAN:FLATness:RESult? (Query Only)

Returns the summary results of each segment of the WLAN Spectral Flatness display.

NOTE: WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

`READ:WLAN:FLATness:RESult?`

Returns

The data is returned as 12 comma delineated text values. The specific values are :

- minimum subcarrier within segment 1
- minimum subcarrier average energy deviation from the global average within segment 1
- deviation the of minimum subcarrier average from the allowed value in segment 1
- maximum subcarrier within segment 1
- maximum subcarrier average energy deviation from the global average within segment 1
- deviation of the maximum subcarrier average from the allowed value in segment 1
- minimum subcarrier within segment 2
- minimum subcarrier average energy deviation from the global average within segment 2
- deviation the of minimum subcarrier average from the allowed value in segment 2

- maximum subcarrier within segment 2
- maximum subcarrier average energy deviation from the global average within segment 2
- deviation of the maximum subcarrier average from the allowed value in segment 2

Examples

`READ:WLAN:FLAT:RES?` might return `-6.0000000000,-17.0626174659E-3,3.9829373825,12.0000000000,19.979260861`, which represents the summary results of each segment of the WLAN Spectral Flatness display.

READ:WLAN:FLATness:TRACe<x>? (Query Only)

Returns the WLAN Spectral Flatness trace data. When <x> is 1, the parameter is Matrix (symbol deviation per subcarrier). When <x> is 2, the parameter is Average Deviation vs Subcarrier.

NOTE: WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

`READ:WLAN:FLATness:TRACe<x>?`

Returns

`#<num_digit><num_byte><data(1)><data(2)>..<data(n)>`

Where:

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`READ:WLAN:FLAT:TRAC2` might return `#510400..` (10400-byte data) for the measurement.

READ:WLAN:MERRor:TRACe<x>? (Query Only)

Returns the trace data for the selected WLAN Magnitude Error trace.

When <x> is 1, the parameter is Matrix.

When <x> is 2, the parameter is Average versus Symbol.

When <x> is 3, the parameter is Average versus Subcarrier

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

`READ:WLAN:MERRor:TRACe<x>?`

Returns

`#<num_digit><num_byte><data(1)><data(2)>..<data(n)>`

Where:

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`READ:WLAN:MERR:TRAC2?` might return `#510400..` (10400-byte data) for the measurement.

READ:WLAN:PERRor:TRACe<x>? (Query Only)

Returns the trace data for the selected WLAN Phase Error trace.

When <x> is 1, the parameter is Matrix.

When <x> is 2, the parameter is Average versus Symbol.

When <x> is 3, the parameter is Average versus Subcarrier

Group

Read commands

Syntax

`READ:WLAN:PERRor:TRACe<x>?`

Returns

`#<num_digit><num_byte><data(1)><data(2)>..<data(n)>`

Where:

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`READ:WLAN:PERR:TRAC2?` might return `#510400..` (10400-byte data) for the measurement.

READ:WLAN:PVTime:BURSt:POWer? (Query Only)

Returns the average power of the burst packet, in dBm, for the WLAN Power vs. Time trace.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

`READ:WLAN:PVTime:BURSt:POWer?`

Returns

Floating point number that represents the average power of the burst packet, in dBm.

Examples

`READ:WLAN:PVT:BURST:POWER?` might return `-1.99`, which represents the value of the average power of the burst packet of -1.99 dBm.

READ:WLAN:PVTime:BURSt:WIDTh? (Query Only)

Returns the value of the measured time width of the burst packet from Power-On to Power-Down (or end of waveform, if that occurs before Power-Down), in seconds, for the WLAN Power vs. Time trace.

Group

Read commands

Syntax

`READ:WLAN:PVTime:BURSt:WIDTh?`

Returns

Floating point number which represents the value of the measured time width of the burst packet from Power-On to Power-Down (or end of waveform, if that occurs before Power-Down), in seconds.

Examples

`READ:WLAN:PVT:BURST:WIDTH?` might return `156.0000000000E-6`, which represents a burst packet width of 156 µS.

READ:WLAN:PVTime:FALL:TIME? (Query Only)

Queries the value of the width of the falling edge, in seconds.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

`READ:WLAN:PVTime:FALL:TIME?`

Returns

Floating point number which represents the value of the width of the falling edge of the burst, in seconds.

Examples

`READ:WLAN:PVT:FALL:TIME?` might return `7e-6`, which indicates the width value of the falling edge is 7 µs.

READ:WLAN:PVTime:FALL:TIME:STARt? (Query Only)

Queries the value of the start time of the falling edge, in seconds.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

`READ:WLAN:PVTime:FALL:TIME:STARt?`

Returns

Floating point number which represents the value of the start time of the falling edge of the burst, in seconds.

Examples

`READ:WLAN:PVT:FALL:TIME:STAR?` might return `6e-6`, which indicates that the start time value of the falling edge is at 6 µs.

READ:WLAN:PVTime:FALL:TIME:STOP? (Query Only)

Queries the value of the stop time of the falling edge, in seconds.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

`READ:WLAN:PVTime:FALL:TIME:STOP?`

Returns

Floating point number which represents the value of the stop time of the falling edge of the burst, in seconds.

Examples

`READ:WLAN:PVT:FALL:TIME:STOP?` might return `7e-6`, which indicates the stop time value of the falling edge is at 7 µs.

READ:WLAN:PVTime:FTIMe? (Query Only)

Queries the value of the width of the falling edge, in seconds.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

`READ:WLAN:PVTime:FTIMe?`

Returns

Floating point number which represents the value of the width of the falling edge of the burst, in seconds.

Examples

`READ:WLAN:PVT:FALL:TIME?` might return `7e-6`, which indicates the width of the falling edge is 7 µs.

READ:WLAN:PVTime:RISE:TIME? (Query Only)

Queries the value of the width of the rising edge, in seconds.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

`READ:WLAN:PVTime:RISE:TIME?`

Returns

Floating point number which represents the value of the width of the rising edge of the burst, in seconds.

Examples

`FETCH:WLAN:PVT:RISE:TIME?` might return `7e-6`, which indicates the width value of the rising edge is 7 µs.

READ:WLAN:PVTime:RISE:TIME:STARt? (Query Only)

Queries the value of the start time of the rising edge, in seconds.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

`READ:WLAN:PVTime:RISE:TIME:STARt?`

Returns

Floating point number which represents the value of the start time of the rising edge of the burst, in seconds.

Examples

`READ:WLAN:PVT:RISE:TIME:STAR?` might return `6e-6`, which indicates the start time value of the rising edge is at 6 µs.

READ:WLAN:PVTime:RISE:TIME:STOP? (Query Only)

Queries the value of the stop time of the rising edge, in seconds.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

`READ:WLAN:PVTime:RISE:TIME:STOP?`

Returns

Floating point number which represents the value of the stop time of the rising edge of the burst, in seconds.

Examples

`FETCH:WLAN:PVT:RISE:TIME:STOP?` might return `7e-6`, which indicates the stop time value of the rising edge is at 7 s.

READ:WLAN:PVTime:RTIMe? (Query Only)

Queries the value of the width of the rising edge, in seconds.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

`READ:WLAN:PVTime:RTIMe?`

Returns

Floating point number which represents the value of the width of the rising edge of the burst, in seconds.

Examples

`READ:WLAN:PVT:RISE:TIME?` might return `7e-6`, which indicates the width value of the rising edge is 7 µs.

READ:WLAN:PVTime:TRACe:X? (Query Only)

Returns the horizontal values (time in seconds) for the WLAN Power vs. Time trace.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

`READ:WLAN:PVTime:TRACe:X?`

Returns

`#<num_digit><num_byte><data(1)><data(2)>..<data(n)>`

Where:

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`READ:WLAN:PVT:TRAC:X` might return `#43204` (3204-byte data), which represents the value of the time in seconds (horizontal values) for the WLAN Power vs. Time trace.

READ:WLAN:PVTime:TRACe:XY? (Query Only)

Returns the horizontal value (time in seconds) and vertical value (power) for the WLAN Power vs. Time trace.

<x(n)><y(n)> is the horizontal value and vertical value pair at the nth data point.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

`READ:WLAN:PVTime:TRACe:XY?`

Returns

`#<num_digit><num_byte><data(1)><data(2)>..<data(n)>`

Where:

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`READ:WLAN:PVT:TRAC:XY?` might return `#43204` (3204-byte data), which represents the horizontal value (time in seconds) and vertical value (power) pair at the nth data point.

READ:WLAN:PVTime:TRACe[:Y]? (Query Only)

Returns the vertical values (power) for the WLAN Power vs. Time trace.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

`READ:WLAN:PVTime:TRACe[:Y]?`

Returns

`#<num_digit><num_byte><data(1)><data(2)>..<data(n)>`

Where:

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

`<data(n)>` is the data value at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples

`FETCH:WLAN:PVT:TRAC:Y?` might return `#43204` (3204-byte data), which represents the vertical values (power) for the WLAN Power vs. Time trace.

READ:WLAN:STABle:VALUe? (Query Only)

Returns the number of bytes in the symbol table for the WLAN measurement.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

`READ:WLAN:STABle:VALUe?`

Returns

`#<num_digit><num_byte>`

Where

`<num_digit>` is the number of digits in `<num_byte>`.

`<num_byte>` is the number of bytes of data that follow.

Examples

`READ:WLAN:STABLE:VALUE?` might return `#43848`, indicating that there are 3848 bytes of data in the symbol table.

READ:WLAN:SUMMary:BURSt:INDex? (Query Only)

Returns the index of the analyzed packet within the analysis record.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

`READ:WLAN:SUMMary:BURSt:INDex?`

Arguments

None

Returns

`<NRf>` data in dB.

READ:WLAN:SUMMary:BURSt:POWer? (Query Only)

Returns the average power of all symbols in the packet, including Preamble and Data segments.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

`READ:WLAN:SUMMary:BURSt:POWer?`

Arguments

None

Returns

`<NRf>` data in dBm.

READ:WLAN:SUMMary:BURSt:POWer:CFACtor? (Query Only)

Returns the peak-to-average burst power factor in the WLAN measurement. The ratio of the highest instantaneous signal power level to the average signal power.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

```
READ:WLAN:SUMMary:BURSt:POWer:CFACtor?
```

Arguments

None

Returns

<NRf> data in dB.

READ:WLAN:SUMMary:CERRor? (Query Only)

Returns the symbol clock error in the WLAN measurement.

Not available for 802.11b.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

```
READ:WLAN:SUMMary:CERRor?
```

Arguments

None

Returns

<NRf> data in parts per million.

READ:WLAN:SUMMary:CPE? (Query Only)

Returns the RMS magnitude error of the pilots over all data symbols.

Not available for 802.11b.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

```
READ:WLAN:SUMMary:CPE?
```

Arguments

None

Returns

<NRf> data in percent.

READ:WLAN:SUMMary:EVM:PEAK:ALL? (Query Only)

Returns the peak EVM value for all subcarriers in the WLAN measurement or for all chips depending on the selected standard.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

```
READ:WLAN:SUMMary:EVM:PEAK:ALL?
```

Arguments

None

Returns

<NRf> data in dB.

READ:WLAN:SUMMary:EVM:PEAK:CHIP:ALL? (Query Only)

Returns the chip at which the reported peak EVM value occurred.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

```
READ:WLAN:SUMMary:EVM:PEAK:CHIP:ALL?
```

Arguments

None

Returns

<NR1>

READ:WLAN:SUMMary:EVM:PEAK:CHIP:ONEKchips? (Query Only)

Returns the chip at which the reported peak EVM value over the first 1000 chips occurred.

Only available for 802.11b.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

```
READ:WLAN:SUMMary:EVM:PEAK:CHIP:ONEKchips?
```

Arguments

None

Returns

<NR1>

READ:WLAN:SUMMary:EVM:PEAK:DATA? (Query Only)

Returns the peak EVM value for the data subcarriers in the WLAN measurement.

Not available for 802.11b.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

```
READ:WLAN:SUMMary:EVM:PEAK:DATA?
```

Arguments

None

Returns

<NRf> data in dB.

READ:WLAN:SUMMary:EVM:PEAK:ONEKchips? (Query Only)

Returns the peak EVM value measured over the first 1000 chips.

Available only for 802.11b.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax
`READ:WLAN:SUMMary:EVM:PEAK:ONEKchips?`

Arguments
None

Returns
`<NRf>` data in dB.

READ:WLAN:SUMMary:EVM:PEAK:PILots? (Query Only)
Returns the peak EVM value for the pilots in the WLAN measurement.

Not available for 802.11b.

Conditions
Measurement view: WLAN

This command requires WLAN Measurements

Group
Read commands

Syntax
READ:WLAN:SUMMary:EVM:PEAK:PILots?

Arguments
None

Returns
`<NRf>` data in dB.

READ:WLAN:SUMMary:EVM:PEAK:SCARrier:ALL? (Query Only)
Returns the subcarrier on which the ALL peak EVM value occurred.

Conditions
Measurement view: WLAN

This command requires WLAN Measurements

Group
Read commands

Syntax
`READ:WLAN:SUMMary:EVM:PEAK:SCARrier:ALL?`

Arguments
None

Returns
`<NR1>`

READ:WLAN:SUMMary:EVM:PEAK:SCARrier:DATA? (Query Only)

Returns the subcarrier on which the DATA peak EVM value occurred.

Not available for 802.11b.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

```
READ:WLAN:SUMMary:EVM:PEAK:SCARrier:DATA?
```

Arguments

None

Returns

<NR1>

READ:WLAN:SUMMary:EVM:PEAK:SCARrier:PILots? (Query Only)

Returns the subcarrier on which the PILOT peak EVM value occurred.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

```
READ:WLAN:SUMMary:EVM:PEAK:SCARrier:PILots?
```

Arguments

None

Returns

<NR1>

READ:WLAN:SUMMary:EVM:PEAK:SYMBol:ALL? (Query Only)

Returns the symbol on which the reported EVM value occurred.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

```
READ:WLAN:SUMMary:EVM:PEAK:SYMBol:ALL?
```

Arguments

None

Returns

<NR1>

READ:WLAN:SUMMary:EVM:PEAK:SYMBol:DATA? (Query Only)

Returns the symbol on which the DATA peak EVM value occurred.

Not available for 802.11b.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

```
READ:WLAN:SUMMary:EVM:PEAK:SYMBol:DATA?
```

Arguments

None

Returns

<NR1>

READ:WLAN:SUMMary:EVM:PEAK:SYMBol:PILots? (Query Only)

Returns the symbol on which the PILOT peak EVM value occurred.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

```
READ:WLAN:SUMMary:EVM:PEAK:SYMBol:PILots?
```

Arguments

None

Returns

<NR1>

READ:WLAN:SUMMary:EVM:RMS:ALL? (Query Only)

Returns the RMS EVM data for all subcarriers in the WLAN measurement and for all chips for 802.11b.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

READ:WLAN:SUMMary:EVM:RMS:ALL?

Arguments

None

Returns

<NRf> RMS EVM in dB

READ:WLAN:SUMMary:EVM:RMS:AVERage? (Query Only)

Returns the average RMS EVM value in the WLAN measurement over the selected number of bursts.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

READ:WLAN:SUMMary:EVM:RMS:AVERage?

Arguments

None

Returns

<NRf> RMS EVM in dB

READ:WLAN:SUMMary:EVM:RMS:DATA? (Query Only)

Returns the RMS EVM data for the data subcarriers in the WLAN measurement.

Not available for 802.11b.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

```
READ:WLAN:SUMMary:EVM:RMS:DATA?
```

Arguments

None

Returns

<NRf> RMS EVM in dB

READ:WLAN:SUMMary:EVM:RMS:MAXimum? (Query Only)

Returns the maximum RMS EVM value in the WLAN measurement over the selected number of bursts.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

```
READ:WLAN:SUMMary:EVM:RMS:MAXimum?
```

Arguments

None

Returns

<NRf> RMS EVM in dB

READ:WLAN:SUMMary:EVM:RMS:ONEKchips? (Query Only)

Returns the RMS EVM data for the 1k chips in the WLAN measurement measured over the first 1000 chips.

Only available for 802.11b.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

```
READ:WLAN:SUMMary:EVM:RMS:ONEKchips?
```

Arguments

None

Returns

<NRf> RMS EVM in dB

READ:WLAN:SUMMary:EVM:RMS:PILots? (Query Only)

Returns the RMS EVM data for the pilot subcarriers in the WLAN measurement.

Not available for 802.11b.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

```
READ:WLAN:SUMMary:EVM:RMS:PILots?
```

Arguments

None

Returns

<NRf> data in percent.

READ:WLAN:SUMMary:FERRor? (Query Only)

Returns the frequency difference between the measured carrier frequency of the signal and the measurement frequency setting.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

```
READ:WLAN:SUMMary:FERRor?
```

Arguments

None

Returns

<NRf> data in kHz.

READ:WLAN:SUMMary:HEADer? (Query Only)

Returns the received packet header information for the WLAN measurement.

Applies to the 802.11b standard only.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

```
READ:WLAN:SUMMary:HEADer?
```

Arguments

None

Returns

Results are returned in the same left-to-right order in which they are presented in the display.

READ:WLAN:SUMMary:HTSig? (Query Only)

Returns the decimal values of the received packet HT-SIGNAL. Applies to the 802.11n standard only.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

```
READ:WLAN:SUMMary:HTSig?
```

Arguments

None

Returns

Results are returned in the same left-to-right order in which they are presented in the display.

READ:WLAN:SUMMary:IQOFfset? (Query Only)

Returns the average magnitude of the DC subcarrier level relative to total signal power.

Group

Read commands

Syntax

```
READ:WLAN:SUMMary:IQOFfset?
```

Arguments

None

Returns

<NRf> data in dB.

READ:WLAN:SUMMary:LSIG? (Query Only)

Returns the decimal values of the received packet (legacy) SIGNAL symbols fields. Applies to the 802.11n standard only.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

```
READ:WLAN:SUMMary:LSIG?
```

Arguments

None

Returns

Results are returned in the same left-to-right order in which they are presented in the display.

READ:WLAN:SUMMary:PACKet? (Query Only)

Returns the type and number of symbols, EVM-RMS and average power of the Preamble and Data portions of the packet.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

```
READ:WLAN:SUMMary:PACKet?
```

Arguments

None

Returns

Results are returned in the same left-to-right order in which they are presented in the display.

READ:WLAN:SUMMary:PACKet:DMODulation? (Query Only)

Returns the modulation used in the Data symbols.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

```
READ:WLAN:SUMMary:PACKet:DMODulation?
```

Arguments

None

Returns

<string> The data modulation being used.

READ:WLAN:SUMMary:PACKet:FORMat? (Query Only)

Returns the packet format setting for the WLAN measurement

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

```
READ:WLAN:SUMMary:PACKet:FORMat?
```

Arguments

None

Returns

<string> The packet format setting.

READ:WLAN:SUMMary:PACKet:GUARd? (Query Only)

Returns the packet guard interval used by the DATA symbols.

Not available for 802.11b.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax
`READ:WLAN:SUMMary:PACKet:GUARd?`

Arguments
None

Returns
`<NRf>` The packet guard interval, 1/4 or 1/8.

READ:WLAN:SUMMary:SIG? (Query Only)
Returns the decimal values of the received packet SIGNAL symbols fields.

Available only for 802.11agjp.

Conditions
Measurement view: WLAN

This command requires WLAN Measurements

Group
Read commands

Syntax
`READ:WLAN:SUMMary:SIG?`

Arguments
None

Returns
Results are returned in the same left-to-right order in which they are presented in the display.

READ:WLAN:SUMMary:VHTSig:A? (Query Only)
Returns the decimal values of the received packet VHT-SIGNAL(A) symbols' fields.

Available only for 802.11ac.

Conditions
Measurement view: WLAN

This command requires WLAN Measurements

Group
Read commands

Syntax
`READ:WLAN:SUMMary:VHTSig:A?`

Arguments
None

Returns

Results are returned in the same left-to-right order in which they are presented in the display.

READ:WLAN:SUMMary:VHTSig:B? (Query Only)

Returns the decimal values of the received packet VHT-SIGNAL(B) symbols fields.

Available only for 802.11ac.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Read commands

Syntax

```
READ:WLAN:SUMMary:VHTSig:B?
```

Arguments

None

Returns

Results are returned in the same left-to-right order in which they are presented in the display.

Sense commands

[SENSe]:ACPower:AVERage

Sets or queries how to average waveform in the Channel power and ACPR measurement.

Conditions

Measurement views: Channel power and ACPR

Group

Sense commands

Syntax

```
[SENSe]:ACPower:AVERage { OFF | TIMe | FREQuency }
[SENSe]:ACPower:AVERage?
```

Arguments

`OFF` disables averaging.

`TIMe` averages waveform using time samples.

`FREQuency` averages waveform using frequency samples.

Examples

`SENSe:ACPOWER:AVERAGE TIMe` averages waveform using time samples.

[SENSe]:ACPower:AVERage:COUNt

Sets or queries the number of traces for averaging in the Channel power and ACPR measurement.

Conditions

Measurement views: Channel power and ACPR

Group

Sense commands

Syntax

```
[SENSe]:ACPower:AVERage:COUNt <number>
[SENSe]:ACPower:AVERage:COUNt?
```

Arguments

`<number>::<NR1>` specifies the average count. Range: 2 to 10000.

Examples

`SENSe:ACPOWER:AVERAGE:COUNT 64` sets the average count to 64.

[SENSe]:ACPower:{BANDwidth|BWIDth}[:RESolution]

Sets or queries the resolution bandwidth (RBW). Programming a specified RBW sets `[SENSe]:ACPower:{BANDwidth|BWIDth}[:RESolution]:AUTO OFF`.

Conditions

Measurement views: Channel power and ACPR

Group

Sense commands

Syntax

```
[SENSe]:ACPower:{BANDwidth|BWIDth}[:RESolution] <value>
[SENSe]:ACPower:{BANDwidth|BWIDth}[:RESolution]?
```

Related Commands

[SENSe]:ACPower:{BANDwidth|BWIDth}[:RESolution]:AUTO (on page 1933)

Arguments

`<value>::<NRf>` specifies the RBW. Range: 100 Hz to 5 MHz.

Examples

`SENSe:ACPOWER:BANDWIDTH:RESOLUTION 200kHz` sets the RBW to 200 kHz.

[SENSe]:ACPower:{BANDwidth|BWIDth}[:RESolution]:ACTual? (Query Only)

Queries the actual resolution bandwidth (RBW) in the Channel power and ACPR measurement.

Conditions

Measurement views: Channel power and ACPR

Group

Sense commands

Syntax

```
[SENSe]:ACPower:{BANDwidth|BWIDth}[:RESolution]:ACTual?
```

Arguments

None

Returns

<NRf> is the actual RBW in Hz.

Examples

SENSe:ACPOWER:BANDWIDTH:RESOLUTION:ACTUAL? might return 299.624E+3, indicating that the actual RBW is 299.624 kHz.

[SENSe]:ACPower:{BANDwidth|BWIDth}[:RESolution]:AUTO

Determines whether to set the resolution bandwidth (RBW) automatically or manually in the Channel power and ACPR measurement.

Conditions

Measurement views: Channel power and ACPR

Group

Sense commands

Syntax

```
[SENSe]:ACPower:{BANDwidth|BWIDth}[:RESolution]:AUTO { OFF | ON | 0 | 1 }
[SENSe]:ACPower:{BANDwidth|BWIDth}[:RESolution]:AUTO?
```

Arguments

OFF or 0 specifies that the RBW is set manually using the [SENSe]:ACPower: {BANDwidth|BWIDth} [:RESolution] command.

ON or 1 specifies that the RBW is set automatically.

Examples

SENSe:ACPOWER:BANDWIDTH:AUTO ON sets the RBW automatically.

[SENSe]:ACPower:{BANDwidth|BWIDth}:VIDeo

Sets or queries the video bandwidth (VBW). Programming a specified VBW sets [SENSe]:ACPower:{BANDwidth| BWIDth}:VIDeo:STATe OFF.

Conditions

Measurement views: Channel power and ACPR

Group

Sense commands

Syntax

```
[SENSe]:ACPower:{BANDwidth|BWIDth}:VIDeo <value>
[SENSe]:ACPower:{BANDwidth|BWIDth}:VIDeo?
```

Related Commands

[SENSe]:ACPower:{BANDwidth|BWIDth}:VIDeo:STATe (on page 1934)

Arguments

`<value>::<NRf>` specifies the VBW.

Range: Current RBW/10^4 (1 Hz minimum) to Current RBW.

Examples

`SENSe:ACPOWER:BANDWIDTH:VIDEO 200kHz` sets the VBW to 200 kHz.

[SENSe]:ACPower:{BANDwidth|BWIDth}:VIDeo:STATe

Determines whether to enable or disable the video bandwidth (VBW) in the Channel power and ACPR measurement.

Conditions

Measurement views: Channel power and ACPR

Group

Sense commands

Syntax

```
[SENSe]:ACPower:{BANDwidth|BWIDth}:VIDeo:STATe { OFF | ON | 0 | 1 }
[SENSe]:ACPower:{BANDwidth|BWIDth}:VIDeo:STATe?
```

Arguments

`OFF` or `0` disables the VBW.

`ON` or `1` enables the VBW.

Examples

`SENSe:ACPOWER:BANDWIDTH:VIDEO:STATE ON` enables the VBW.

[SENSe]:ACPower:CHANnel:{BANDwidth|BWIDth}

Sets or queries frequency bandwidth of each channel (all share the same value) in the Channel power and ACPR measurement.

Conditions

Measurement views: Channel power and ACPR

Group

Sense commands

Syntax

```
[SENSe]:ACPower:CHANnel:{BANDwidth|BWIDth} <value>
[SENSe]:ACPower:CHANnel:{BANDwidth|BWIDth}?
```

Arguments

`<value>::<NRf>` specifies the channel bandwidth. Range: 1 Hz to full span.

Examples

`SENSe:ACPOWER:CHANNEL:BANDWIDTH 1.5MHz` sets the channel bandwidth to 1.5 MHz.

[SENSe]:ACPower:CHANnel:FILTer

Sets or queries the adjacent channel filter in the Channel power and ACPR measurement.

Conditions

Measurement views: Channel power and ACPR

Group

Sense commands

Syntax

```
[SENSe]:ACPower:CHANnel:FILTer { RRCosine | NONE }
[SENSe]:ACPower:CHANnel:FILTer?
```

Arguments

`RRCosine` uses the Root-Raised Cosine filter.

`NONE` uses no filter.

Examples

`SENSe:ACPOWER:CHANNEL:FILTER RRCosine` uses the Root-Raised Cosine filter for the Channel power and ACPR measurement.

[SENSe]:ACPower:CHANnel:PAIRs

Sets or queries the number of adjacent channel pairs (upper and lower) in the Channel power and ACPR measurement.

Conditions

Measurement views: Channel power and ACPR

Group

Sense commands

Syntax

```
[SENSe]:ACPower:CHANnel:PAIRs <number>
[SENSe]:ACPower:CHANnel:PAIRs?
```

Arguments

`<number>::<NR1>` specifies the number of adjacent pairs. Range: 0 to 50.

Examples

`SENSe:ACPOWER:CHANNEL:PAIRS 5` sets five adjacent channel pairs.

[SENSe]:ACPower:CHANnel:SPACing

Sets or queries frequency difference between centers of each channel in the Channel power and ACPR measurement.

Conditions

Measurement views: Channel power and ACPR

Group

Sense commands

Syntax

```
[SENSe]:ACPower:CHANnel:SPACing <value>
[SENSe]:ACPower:CHANnel:SPACing?
```

Arguments

`<value>::<NRf>` specifies the channel-to-channel spacing.

Range: 1 Hz to 1 GHz.

Examples

`SENSe:ACPOWER:CHANNEL:SPACING 5MHz` sets the channel-to-channel spacing to 5 MHz.

[SENSe]:ACPower:CHIPrate

Sets or queries the chip rate when `[SENSe]:ACPower:CHANnel:FILTer` is set to RRCosine (Root Raised Cosine).

Conditions

Measurement views: Channel power and ACPR

Group

Sense commands

Syntax

```
[SENSe]:ACPower:CHIPrate <value>
[SENSe]:ACPower:CHIPrate?
```

Related Commands

[SENSe]:ACPower:CHANnel:FILTer (on page 1935)

Arguments

`<value>::<NRf>` specifies the chip rate. Range: 100 Hz to 105 MHz.

Examples

`SENSe:ACPOWER:CHIPRATE 5kHz` sets the chip rate to 5 kHz.

[SENSe]:ACPower:CLEar:RESults (No Query Form)

Restarts the average trace.

Conditions

Measurement views: Channel power and ACPR

Group

Sense commands

Syntax

```
[SENSe]:ACPower:CLEar:RESults
```

Arguments

None

Examples

`SENSe:ACPOWER:CLEAR:RESULTS` restarts the average trace.

[SENSe]:ACPower:FREQuency

Sets or queries the center frequency in the Channel power and ACPR measurement.

Conditions

Measurement views: Channel power and ACPR

Group

Sense commands

Syntax

```
[SENSe]:ACPower:FREQuency <value>
[SENSe]:ACPower:FREQuency?
```

Arguments

<value>::<NRf> specifies the center frequency.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples

`SENSe:ACPOWER:FREQUENCY 2.35GHz` sets the center frequency to 2.35 GHz.

[SENSe]:ACPower:FREQuency:STEP

Sets or queries the frequency step size in the Channel power and ACPR measurement. Programming a specified step size sets `[SENSe]:ACPower:FREQuency:STEP:AUTO OFF`.

Conditions

Measurement views: Channel power and ACPR

Group

Sense commands

Syntax

```
[SENSe]:ACPower:FREQuency:STEP <value>
[SENSe]:ACPower:FREQuency:STEP?
```

Related Commands

[SENSe]:ACPower:FREQuency:STEP:AUTO (on page 1938)

Arguments

`<value>::<NRf>` specifies the frequency step size.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples

`SENSe:ACPOWER:FREQUENCY:STEP 1kHz` sets the frequency step size to 1 kHz.

[SENSe]:ACPower:FREQuency:STEP:AUTO

Determines whether to set the frequency step size automatically or manually in the Channel power and ACPR measurement.

Conditions

Measurement views: Channel power and ACPR

Group

Sense commands

Syntax

```
[SENSe]:ACPower:FREQuency:STEP:AUTO { OFF | ON | 0 | 1 }
[SENSe]:ACPower:FREQuency:STEP:AUTO?
```

Arguments

`OFF` or `0` specifies that the frequency step size is set manually using the `[SENSe]:ACPower:FREQuency:STEP` command.

`ON` or `1` specifies that the frequency step size is set automatically.

Examples

`SENSe:ACPOWER:FREQUENCY:STEP:AUTO ON` specifies that the frequency step size is set automatically.

[SENSe]:ACPower:NFLoor:STATe

Determines whether to enable or disable the correction for noise floor.

Conditions

Measurement views: Channel power and ACPR

Group

Sense commands

Syntax

```
[SENSe]:ACPower:NFLoor:STATe { OFF | ON | 0 | 1 }
[SENSe]:ACPower:NFLoor:STATe?
```

Arguments

`OFF` or `0` disables the correction for noise floor.

`ON` or `1` enables the correction for noise floor.

Examples

`SENSe:ACPOWER:NFLOOR:STATE ON` enables the correction for noise floor.

[SENSe]:ACPower:POINts:COUNt

Sets or queries the number of points used for the trace in the Channel Power and ACPR measurement.

Conditions

Measurement views: Channel power and ACPR

Group

Sense commands

Syntax

```
[SENSe]:ACPower:POINts:COUNt { P801 | P2401 | P4001 | P8001 | P10801 }
[SENSe]:ACPower:POINts:COUNt?
```

Arguments

`RTBandwidth` optimizes the measurement for real-time bandwidth.

`DRANge` optimizes the measurement for dynamic range.

Examples

`SENSE:ACPOWER:POINTS:COUNT P4001` sets the number of points used in the trace of the Channel Power and ACPR display to 4001.

[SENSe]:ACPower:RRCRolloff

Sets or queries the filter parameter (roll-off ratio) for the Root Raised Cosine filter.

Conditions

Measurement views: Channel power and ACPR

Group

Sense commands

Syntax

```
[SENSe]:ACPower:RRCRolloff <value>
[SENSe]:ACPower:RRCRolloff?
```

Related Commands

[SENSe]:ACPower:CHANnel:FILTer (on page 1935)

Arguments

`<value>::<NRf>` specifies the filter parameter.

Range: 0.0001 to 1 in 0.0001 steps.

Examples

`SENSe:ACPOWER:RRCROLLOFF 0.3` sets the filter parameter to 0.3.

[SENSe]:ACPower:SYMBrate

Sets or queries the symbol rate when `[SENSe]:ACPower:CHANnel:FILTer` is set to RRCosine (Root Raised Cosine).

Conditions

Measurement views: Channel power and ACPR

Group

Sense commands

Syntax

```
[SENSe]:ACPower:SYMBrate <value>
[SENSe]:ACPower:SYMBrate?
```

Related Commands

[SENSe]:ACPower:CHANnel:FILTer (on page 1935)

Arguments

`<value>::<NRf>` specifies the symbol rate. Range: 100 Hz to 105 MHz.

Examples

`SENSe:ACPOWER:CHIPRATE 5kHz` sets the symbol rate to 5 kHz.

[SENSe]:ACQuisition:{BANDwidth|BWIDth}

Sets or queries the acquisition bandwidth (frequency range of the acquisition) when `[SENSe]:ACQuisition:MODE` is set to SAMPles or LENGth.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:ACQuisition:{BANDwidth|BWIDth} <value>
[SENSe]:ACQuisition:{BANDwidth|BWIDth}?
```

Arguments

`<value>::<NRf>` specifies the acquisition bandwidth.

Range: 1MHz to 40 MHz (Standard) / 110MHz (Option 110).

Examples

`SENSe:ACQUISITION:BANDWIDTH 30MHz` sets the acquisition bandwidth to 30 MHz.

[SENSe]:ACQuisition:FFRame:ACTual? (Query Only)

Queries the actual number of Fast Frames.

Conditions

Measurement views: All

Group

Sense commands

Syntax

`[SENSe]:ACQuisition:FFRame:ACTual?`

Arguments

None

Returns

`<NR1>` The actual number of Fast Frames.

Examples

`SENSe:ACQUISITION:FFRAME:ACTUAL?` might return 178, indicating that the actual number of Fast Frames is 178.

[SENSe]:ACQuisition:FFRame:LIMit

Sets or queries the limit number of Fast Frames.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:ACQuisition:FFRame:LIMit
[SENSe]:ACQuisition:FFRame:LIMit?
```

Arguments

`<value>::<NR1>` specifies the fast frame limit. Range: 1 to 65535 frames.

Where 1 to 65535 frames for RSA 5000A series. For others, the range is 1 to 220 -1 frames.

Examples

`SENSe:ACQUISITION:FFRAME:LIMIT 500` sets the Fast Frame limit to 500.

[SENSe]:ACQuisition:FFRame:STATe

Determines whether to enable or disable the Fast Frame.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:ACQuisition:FFRame:STATe { OFF | ON | 0 | 1 }
[SENSe]:ACQuisition:FFRame:STATe?
```

Arguments

`OFF` or `0` disables the Fast Frame.

`ON` or `1` enables the Fast Frame.

Examples

`SENSe:ACQUISITION:FFRAME:STATE ON` enables the Fast Frame.

[SENSe]:ACQuisition:FSAVe:ENAble

Set or queries the FastSave acquisition mode.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:ACQuisition:FSAVe:ENAble { OFF | ON | 0 | 1 }
[SENSe]:ACQuisition:FSAVe:ENAble?
```

Arguments

`OFF` or `0` disables the FastSave acquisition.

`ON` or `1` enables the FastSave acquisition..

Examples

`SENSe:ACQUISITION:FSAVE:ENABLE ON` enables the FastSave acquisition.

[SENSe]:ACQuisition:FSAVe:FILEs:MAXimum

Set or queries the FastSave maximum number of files saved per run.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:ACQuisition:FSAVe:FILEs:MAXimum <value>
```

[SENSe]:ACQuisition:FSAVe:FILEs:MAXimum?

Arguments

<value>::<NR2> specifies the maximum number of files saved per run. Range: 1 to 1,000,000 files.

NOTE: When the FILEs:MAXimum is reached, an event is added to the event queue to indicate that saves are completed. Note that this same event is issued now when Trigger Actions Max total files is reached as well.

Examples

SENSe:ACQUISITION:FSAVE:FILES:MAXIMUM 10 sets the maximum number of files saved per run to 10.

[SENSe]:ACQuisition:FSAVe:FORMat

Set or queries the FastSave file format.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:ACQuisition:FSAVe:FORMat { TIQ| MAT }
[SENSe]:ACQuisition:FSAVe:FORMat?
```

Arguments

TIQ selects the TIQ file format for FastSave acquisitions.

MAT selects the MAT file format for FastSave acquisitions.

Examples

SENSe:ACQUISITION:FSAVE:FORMAT MAT selects the Matlab file format.

[SENSe]:ACQuisition:FSAVe:LOCation

Set or queries the FastSave file save location (a directory).

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:ACQuisition:FSAVe:LOCation <path>
[SENSe]:ACQuisition:FSAVe:LOCation?
```

Arguments

<path>::<string> specifies the location where FastSave acquisitions will be saved.

Examples

`SENSe:ACQUISITION:FSAVE:LOCATION "C:\RSA5100B Files\Test_folder"` sets location where FastSave acquisitions are saved to C:\RSA5100B Files\Test_folder.

[SENSe]:ACQuisition:FSAVe:NAME:BASE

Set or queries the FastSave Base file name.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:ACQuisition:FSAVe:NAME:BASE <file_name>
[SENSe]:ACQuisition:FSAVe:NAME:BASE?
```

Arguments

`<file_name>::<string>` specifies the base file name use for FastSave acquisitions.

Examples

`SENSe:ACQUISITION:FSAVE:NAME:BASE "FSAVE"` sets the Base file name to FSAVE.

[SENSe]:ACQuisition:MEMory:AVAilable:SAMPles? (Query Only)

Returns the amount of acquisition memory available in the instrument.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:ACQuisition:MEMory:AVAilable:SAMPles?
```

Arguments

None

Returns

`<NRf>` The amount of acquisition memory available in samples.

Examples

`SENSe:ACQUISITION:MEMORY:AVAILABLE:SAMPLES?` might return 999.424E+3, indicating that 999424 samples are available.

[SENSe]:ACQuisition:MEMory:CAPacity[:TIME]? (Query Only)

Returns the acquisition memory capacity (maximum period of time that can be acquired with the acquisition memory).

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:ACQuisition:MEMory:CAPacity[:TIME]?
```

Arguments

None

Returns

<NRf> The acquisition memory capacity in seconds.

Examples

SENSe:ACQUISITION:MEMORY:CAPACITY:TIME? might return 26.651E-3, indicating that 26.651 ms can be acquired.

[SENSe]:ACQuisition:MEMory:USED[:PERCent]? (Query Only)

Returns the percentage of the capacity used based on the current settings.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:ACQuisition:MEMory:USED[:PERCent]?
```

Arguments

None

Returns

<NRf> The percentage of the capacity used.

Examples

SENSe:ACQUISITION:MEMORY:USED:PERCENT? might return 50.0, indicating that 50% is used.

[SENSe]:ACQuisition:MODE

Sets or queries the acquisition mode (how to determine the sampling parameters of acquisition bandwidth, samples, and length).

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:ACQuisition:MODE { AUTO | SAMPles | LENGth }
[SENSe]:ACQuisition:MODE?
```

Related Commands

[SENSe]:ACQuisition:{BANDwidth|BWIDth} (on page 1940)

[SENSe]:ACQuisition:SAMPles (on page 1952)

[SENSe]:ACQuisition:SEConds (on page 1953)

Arguments

`AUTO` sets the all sampling parameters automatically.

`SAMPles` sets the acquisition bandwidth and samples manually, using the `[SENSe]:ACQuisition:{BANDwidth|BWIDth}` and :SAMPles commands.

`LENGth` sets the acquisition bandwidth and length manually, using the `[SENSe]:ACQuisition:{BANDwidth|BWIDth}` and :SEConds commands.

Examples

`SENSe:ACQUISITION:MODE AUTO` sets the all sampling parameters automatically.

[SENSe]:ACQuisition:RECord:ACQuisition:{BANDwidth|BWIDth}:AUTO

Set or queries whether the recording acquisition bandwidth is automatic or manual.

Group

Sense commands

Syntax

```
[SENSe]:ACQuisition:RECord:ACQuisition:{BANDwidth|BWIDth}: AUTO { OFF | ON | 0 | 1 }
[SENSe]:ACQuisition:RECord:ACQuisition:{BANDwidth|BWIDth}: AUTO ?
```

Arguments

`ON` or `1` indicates that the acquisition bandwidth during recording is based on the current system acquisition bandwidth.

`OFF` or `0` indicates that the acquisition bandwidth during recording is the selected bandwidth.

Returns

See Arguments.

Examples

`SENSe:ACQUISITION:RECORD:ACQUISITION:BANDWIDTH:AUTO?` might return `0`, indicating that the acquisition bandwidth during recording is the selected bandwidth.

[SENSe]:ACQuisition:RECord:ACQuisition:{BANDwidth|BWIDth} <value>

Set or queries the Acquisition Bandwidth used during recording.

Group

Sense commands

Syntax

```
[SENSe]:ACQuisition:RECord:ACQuisition:{BANDwidth|BWIDth} <value>
[SENSe]:ACQuisition:RECord:ACQuisition:{BANDwidth|BWIDth} <value> ?
```

Arguments

`<value>::<NRf>` specifies the acquisition bandwidth used during recording.

Returns

`<NRf>`

Examples

`SENSe:ACQuisition:RECord:ACQuisition:BWIDth 20` sets the acquisition bandwidth to 20 MHz.

[SENSe]:ACQuisition:RECord:FILE:LENGth

Sets or queries how much data (time in milliseconds, seconds, or minutes) is recorded when recording starts. File Length uses the Max saved files per run: setting to determine how many data files are captured per each recording session.

Conditions

Requires an RSA be connected with SignalVu-PC.

Group

Sense commands

Syntax

```
[SENSe]:ACQuisition:RECord:FILE:LENGth <numeric>
[SENSe]:ACQuisition:RECord:FILE:LENGth?
```

Arguments

`<numeric>::=<NR3>` determines the file length of each data file saved.

Range: 1E-21 to 10E3 when units set to milliseconds

Range: 1E-5 to 10E3 when units set to seconds

Range: 1E-13 to 10E3 when units set to minutes

Returns

Examples

`SENSe:ACQUISITION:RECORD:FILE:LENGTH 10E2` sets the file length to 1000.

[SENSe]:ACQuisition:RECord:FILE:LENGth:ENABle

Sets or queries the file length conditions to determine how much data is recorded. File Length uses theMax saved files per run: setting to determine how many data files are captured per each recording session.

Conditions

Requires an RSA be connected with SignalVu-PC.

Group

Sense commands

Syntax

```
[SENSe]:ACQuisition:RECord:FILE:LENGth:ENABle { OFF | ON | 0 | 1 }
[SENSe]:ACQuisition:RECord:FILE:LENGth:ENABle?
```

Arguments

`ON` or `1` enables File Length.

`OFF` or `0` disables File Length.

Examples

`SENSe:ACQUISITION:RECORD:FILE:LENGTH:ENABLE ON` enables the File Length conditions.

[SENSe]:ACQuisition:RECord:FILE:LENGth:UNITs

Sets or queries the units used for the file length setting.

Conditions

Requires an RSA be connected with SignalVu-PC.

Group

Sense commands

Syntax

```
[SENSe]:ACQuisition:RECord:FILE:LENGth:UNITs { MSEC | SEC | MIN }
```

Arguments

`MSEC`: milliseconds

`SEC`: seconds

`MIN`: minutes

Examples

`SENSe:ACQUISITION:RECORD:FILE:LENGTH:UNITS MSEC` sets the units for the file length to milliseconds.

[SENSe]:ACQuisition:RECord:FILE:LOCation

Sets the file location of the saved data records. To fully support data streaming, the drive selected for recording data must have a minimum write speed of 300 MB/sec.

Conditions

Requires an RSA be connected with SignalVu-PC.

Group

Sense commands

Syntax

```
[SENSe]:ACQuisition:RECord:FILE:LOCation <file_path>
[SENSe]:ACQuisition:RECord:FILE:LOCation?
```

Arguments

<file_path>::=<"string"> specifies the location.

If you omit the directory path, the default path is used, which is C:\SignalVu-PC Files\Sample Data Records initially.

Examples

SENSe:ACQUISITION:RECORD:FILE:LOCATION "C:\SignalVu-PC Files\Sample Data Records" sets the location of the data records to that location.

[SENSe]:ACQuisition:RECord:FILE:MAXimum

Sets or queries how many times a record is created, of the specified File Length setting. If the File Length is not enabled, this setting is ignored. See the command [:SENSe]:ACQuisition:RECord:FILE:LENGth:ENABle.

Conditions

Requires an RSA be connected with SignalVu-PC.

Group

Sense commands

Syntax

```
[SENSe]:ACQuisition:RECord:FILE:MAXimum <value>
[SENSe]:ACQuisition:RECord:FILE:MAXimum?
```

Arguments

<value>::=<NR1> the number of records created.

Range: 1 to 1000000.

Examples

SENSe:ACQUISITION:RECORD:FILE:MAXIMUM 10 sets the number of files to saved to ten.

[SENSe]:ACQuisition:RECord:FILE:NAME:BASE

Sets or queries the base file name that is used for all data recordings. The base name is appended with additional information, creating unique file names.

Conditions

Requires an RSA be connected with SignalVu-PC.

Group

Sense commands

Syntax

```
[SENSe]:ACQuisition:RECord:FILE:NAME:BASE <base_name>
[SENSe]:ACQuisition:RECord:FILE:NAME:BASE?
```

Arguments

<base_name>::=<"string"> specifies the base file name used for all data recordings.

Examples

SENSe:ACQUISITION:RECORD:FILE:NAME:BASE "DataFile" sets the base name to DataFile.

[SENSe]:ACQuisition:RECord:FILE:SIZE? (Query Only)

Queries the expected recording file size in bytes. This item is only displayed in the UI when Record Time is enabled.

Group

Sense commands

Syntax

`[SENSe]:ACQuisition:RECord:FILE:SIZE?`

Arguments

None

Returns

`<NRf>` in mseconds.

Examples

`SENSe:ACQUISITION:RECORD:FILE:SIZE?` might return `10`, indicating that the recorded file size is expected to be 10 ms.

[SENSe]:ACQuisition:RECord:FILE:STRucture

Sets or queries the file structure to use for the recorded data files.

Conditions

Requires an RSA 306, 306B, 500A series, 600A series, of 7100A be connected.

Group

Sense commands

Syntax

`[SENSe]:ACQuisition:RECord:FILE:STRucture { RAW | FORM }`

Arguments

`RAW`: Unformatted files are created. Two files are created for each recording, a header file (.r3h) and the data file (.r3a).

`FORMatted`: Formatted files are created, ending with a .r3f suffix.

MDSCombined: Midas 2.0 (CDIF) files are created.

MDSDetached: Midas 2.0 Detached (CDIF+DET) files are created.

The contents of these file types is described in the API Reference manual (077-1031-XX) available on the SignalVu-PC Software and Documentation Flash Drive or on tek.com.

Examples

`SENSe:ACQUISITION:RECORD:FILE:STRUCTURE FORMATTED` sets the file format to Formatted.

[SENSe]:ACQuisition:RECord:RESults? (Query Only)

Queries the recording status.

Group

Sense commands

Syntax

`[SENSe]:ACQuisition:RECord:RESults?`

Arguments

None

Returns

`<Number files completed>,<bytes in file>,<percentage in file>,<Acquisition Data Status>`

Where

`<Number files completed>::<NR1>` Number of files completed.

`<bytes in file>::<NR1>` Number of bytes written to current file.

`<percentage in file>::<NRf>` Percentage of current file written.

`<Acquisition Data Status>::<NR1>` Acquisition data status. This is the overall acquisition data status since recording started.

The following hexadecimal values indicate acquisition data status.

Description	Value
ADC Overrange	0x1
Acquisition Data Lost	0x20
IF Power Overrange	0x100000

Examples

`SENSe:ACQUISITION:RECORD:RESULTS?` might return `2,2400,50,0x1` indicating that 2 files were completed, 2400 total bytes, 50% completed, and that there is ADC overrange.

[SENSe]:ACQuisition:RECord:STARt (No Query Form)

Starts the data recording. The Record to disk must the set to Record now. See the command `[:SENSe]:ACQuisition:RECord:TYPE`. See the command `[:SENSe]:ACQuisition:RECord:FILE:LOCation` to specify the file name and location.

Conditions

Requires an RSA be connected with SignalVu-PC.

Group

Sense commands

Syntax

`[SENSe]:ACQuisition:RECord:STARt`

Arguments

None

Examples

SENSe:ACQUISITION:RECORD:START starts the data recording.

[SENSe]:ACQuisition:RECord:STOP (No Query Form)

Starts the data recording. The Record to disk must the set to Record now. See the command [:SENSe]:ACQuisition:RECord:TYPE. See the command [:SENSe]:ACQuisition:RECord:FILE:LOCation to specify the file name and location.

Conditions

Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group

Sense commands

Syntax

[SENSe]:ACQuisition:RECord:STOP

Arguments

None

Examples

SENSe:ACQUISITION:RECORD:STOP stops the data recording.

[SENSe]:ACQuisition:RECord:TYPE

Sets or queries when the recording starts streaming data to the specified file.

Conditions

Requires an RSA be connected with SignalVu-PC.

Group

Sense commands

Syntax
[SENSe]:ACQuisition:RECord:TYPE { NOW | TRIGger }

Arguments

NOW starts the recording immediately when Start is enabled. See the command [:SENSe]:ACQuisition:RECord:STARt (on page 1951).

TRIGger starts the recording when an appropriate trigger signal is received and Triggering is set to Triggered. See the command :TRIGger[:SEQuence][:STATus] (on page 2732).

Examples

SENSe:ACQUISITION:RECORD:TYPE NOW sets the Record to disk setting to Record now.

[SENSe]:ACQuisition:SAMPles

Sets or queries the acquisition samples (number of samples acquired over the acquisition time) when [SENSe]:ACQuisition:MODE is set to SAMPles.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:ACQuisition:SAMPles <value>
[SENSe]:ACQuisition:SAMPles?
```

Arguments

`<value>::<NR1>` specifies the acquisition samples. Range: 2 to 1 G samples..

Examples

`SENSe:ACQUISITION:SAMPLES 1114` sets the acquisition samples to 1114.

[SENSe]:ACQuisition:SEConds

Sets or queries the acquisition length (time over which the acquisition occurs) when `[SENSe]:ACQuisition:MODE` is set to LENGth.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:ACQuisition:SEConds <value>
[SENSe]:ACQuisition:SEConds?
```

Arguments

`<value>::<NRf>` specifies the acquisition length.

Examples

`SENSe:ACQUISITION:SAMPLES 12.5ms` sets the acquisition length to `12.5 ms`.

[SENSe]:{AM|FM|PM}:{BANDwidth|BWIDth}:MEASurement

Sets measurement bandwidth for the AM, FM, or PM demodulation to analyze.

Conditions

Measurement views: AM, FM, PM

Group

Sense commands

Syntax

```
[SENSe]:{AM|FM|PM}:{BANDwidth|BWIDth}:MEASurement <value>
[SENSe]:{AM|FM|PM}:{BANDwidth|BWIDth}:MEASurement?
```

Related Commands

FETCh:{AM|FM|PM}? (on page 1062)

READ:AM:RESult? (on page 1557)

READ:FM:RESult? (on page 1652)

READ:PM:RESult? (on page 1758)

Arguments

`<value>::<NRf>` specifies the demodulation bandwidth for the specified modulation type.

Examples

`SENSe:AM:BANDwidth:MEASurement 1e6` sets the amplitude demodulation bandwidth to `1 MHz`.

[SENSe]:{AM|FM|PM}:CLEar:RESults (No Query Form)

Clear results in the AM/FM/PM measurement.

Conditions

Measurement views: AM, FM, PM

Group

Sense commands

Syntax

```
[SENSe]:{AM|FM|PM}:CLEar:RESults
```

Arguments

None

Examples

`SENSe:AM:CLEAR:RESULTS` clears results in the AM measurement.

[SENSe]:{AM|FM|PM}:{MTPoints|MAXTracepoints}

Sets or queries the maximum trace points in the AM/FM/PM measurement.

Conditions

Measurement views: AM, FM, PM

Group

Sense commands

Syntax

```
[SENSe]:{AM|FM|PM}:{MTPoints|MAXTracepoints} { ONEK | TENK | HUNDredk | NDECimate | NEVerdecimate }
[SENSe]:{AM|FM|PM}:{MTPoints|MAXTracepoints}?
```

Arguments

The following table lists the arguments.

Argument	Maximum trace points
ONEK	1k
TENK	10k
HUNDredk	100k
NDECimate or NEVerdecimate	Never decimate

Examples

`SENSe:AM:MTPOINTS ONEK` selects the maximum trace points of 1000 in the AM measurement.

[SENSe]:AM:DETect:AMPLitude

Sets or queries the carrier amplitude detection method used to determine the 0% reference modulation in the AM measurement.

Conditions

Measurement views: AM

Group

Sense commands

Syntax

```
[SENSe]:AM:DETect:AMPLitude { AVERage | MEDian }
[SENSe]:AM:DETect:AMPLitude?
```

Arguments

`AVERage` defines the 0% reference modulation as the average amplitude in the analysis range (default).

`MEDian` defines the 0% reference modulation as the median amplitude ([(maximum)+(minimum)]/2) in the analysis range.

Examples

`SENSe:AM:DETECT:AMPLITUDE AVERage` defines the 0% modulation as the average amplitude in the analysis range.

[SENSe]:ANALysis:ADVanced:DITHer

Determines whether to enable or disable dithering, or set it automatically.

Dither is a random low-level signal consisting of white noise of one quantizing level peak-to-peak amplitude which may be added to an analog signal prior to sampling for the purpose of minimizing quantization error.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:ANALysis:ADVanced:DITHer { AUTO | ON | OFF }
[SENSe]:ANALysis:ADVanced:DITHer?
```

Arguments

AUTO specifies that the dither is set automatically.

ON enables dithering.

OFF disables dithering.

Examples

SENSe:ANALYSIS:ADVANCED:DITHER ON enables dithering.

[SENSe]:ANALysis:ADVanced:DITHer:HWARe:STATus? (Query Only)

Returns the dithering hardware status.

Conditions

Measurement views: All

Group

Sense commands

Syntax

[SENSe]:ANALysis:ADVanced:DITHer:HWARe:STATus?

Arguments

None

Returns

One of the following status information.

Dithering status

Status	Description
DUNaligned	Dithering is disabled and unaligned.
ON	Dithering is enabled.
OFF	Dithering is disabled.

Examples

SENSE:ANALYSIS:ADVANCED:DITHER:HWARE:STATUS? might return OFF, indicating that the dithering is disabled.

[:SENSe]:ANALysis:BANDwidth:FILTer:MODE {set and query}

Sets or queries the measurement bandwidth filter mode.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
SENSe:ANAlysis:BANDwidth:FILTer:MODE {0|1|TRUE|FALSE}
SENSe:ANAlysis:BANDwidth:FILTer:MODE? (Query)
```

Returns

`0` is used to change the 'Measurement Bandwidth Filter' checkbox state from unchecked to checked.

`1` is used to change 'Measurement Bandwidth Filter' checkbox state from checked to unchecked.

Examples

`SENSe:ANAlysis:BANDwidth:FILTer:MODE 0` changes 'Measurement Bandwidth Filter' checkbox state to unchecked.

[SENSe]:ANALysis:LENGth[:SOURce<z>]

Sets or queries the analysis length. Programming a specified length sets `[SENSe]:ANALysis:AUTO OFF`.

The parameter <z> = 1 to number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Default source is Source 1.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:ANALysis:LENGth[:SOURce<z>] <value>
[SENSe]:ANALysis:LENGth[:SOURce<z>]?
```

Related Commands

[SENSe]:ANALysis:LENGth[:SOURce2]:AUTO (on page 1958)

Arguments

`<value>::<NRf>` specifies the analysis length.

Range: 10 ns to [(acquisition length) - 400 ns].

If [(analysis start) + (analysis length)] > [(acquisition length) - 400 ns], the actual analysis length is reduced to [(acquisition length) - 200 ns].

Examples

`SENSe:ANALysis:LENGth:SOURce2 25.625 µs` sets the analysis length of Source 2 to 25.625 µs.

[SENSe]:ANALysis:LENGth[:SOURce<z>]:ACTual? (Query only)

Queries the actual analysis length.

The parameter <z> = 1 to number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Default source is Source 1.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:ANALysis:LENGth[:SOURce<z>]:ACTual?
```

Returns

<NRf> Actual analysis length in seconds.

Examples

SENSe:ANALysis:LENGth:SOURce2:ACTual? might return 25.625E-6, indicating that the actual analysis length is 25.625 µs.

[SENSe]:ANALysis:LENGth[:SOURce<z>]:AUTO

Determines whether to set the analysis length automatically or manually.

The parameter <z> = 1 to number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Default source is Source 1.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:ANALysis:LENGth:SOURce<z>:AUTO { OFF | ON | 0 | 1 }
[SENSe]:ANALysis:LENGth:SOURce<z>:AUTO?
```

Arguments

OFF or 0 sets the analysis length manually, using the [SENSe]:ANALysis:LENGth command.

ON or 1 sets the analysis length automatically.

Examples

SENSe:ANALysis:LENGth:SOURce2:AUTO ON sets the analysis length of Source 2 automatically.

[SENSe]:ANALysis:REFerence[:SOURce<z>]

Sets or queries the analysis time reference.

The parameter <z> = 1 to number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Default source is Source 1.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:ANALysis:REFerence:SOURCE<z> { ACQSTART | TRIGGER }
[SENSe]:ANALysis:REFerence:SOURCE<z>?
```

Arguments

`ACQSTART` specifies the acquisition start as the time zero reference.

`TRIGGER` specifies the trigger point as the time zero reference.

Examples

`SENSe:ANALysis:REFerence:SOURCE2 ACQSTART` specifies the acquisition start as the analysis time reference for Source 2.

[SENSe]:ANALysis:STARt[:SOURce<z>]

Sets or queries the analysis offset time. Programming a specified offset time sets `[SENSe]:ANALysis:STARt:AUTO OFF`.

The parameter <z> = 1 to number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Default source is Source 1.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:ANALysis:STARt:SOURCE<z> <value>
[SENSe]:ANALysis:STARt:SOURCE<z>?
```

Related Commands

[SENSe]:ANALysis:LENGth[:SOURCE<z>] (on page 1958)

[SENSe]:ANALysis:STARt[:SOURCE<z>]:AUTO (on page 1959)

Arguments

`<value>::<NRf>` specifies the analysis offset time.

Range: 0 to [(acquisition length) - 200 ns].

If [(analysis start) + (analysis length)] > [(acquisition length) - 400 ns], the actual analysis length is reduced to [(acquisition length) - 200 ns].

Examples

`SENSe:ANALysis:STARt:SOURCE2 23.5µs` sets the analysis offset to 23.5 µs.

[SENSe]:ANALysis:STARt[:SOURce<z>]:AUTO

Determines whether to set the analysis offset automatically or manually.

The parameter <z> = 1 to number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Default source is Source 1.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:ANALysis:STARt:SOURCE<z>:AUTO { OFF | ON | 0 | 1 }
[SENSe]:ANALysis:SOURCE<z>:STARt:AUTO?
```

Arguments

`OFF` or `0` sets the analysis offset manually, using the `[SENSe]:ANALysis:STARt` command.

`ON` or `1` sets the analysis offset automatically.

Examples

`SENSe:ANALysis:STARt:SOURCE1:AUTO ON` sets the analysis offset automatically.

[SENSe]:ANSPectrum:STARt:AUTO:METHod

Sets or queries the method used for computing auto analysis and spectrum offsets when `[SENSe]:ANALysis:REFerence` is set to TRIGger.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:ANSPectrum:STARt:AUTO:METHod { INCLude | LEGacy }
[SENSe]:ANSPectrum:STARt:AUTO:METHod?
```

Arguments

`INCLude` includes the trigger point.

`LEGacy` starts a trigger point.

Examples

`SENSe:ANSPECTRUM:START:AUTO:METHOD INCLUDE` includes the trigger point.

[SENSe]:AUDio:AVERage:COUNt

Sets or queries the number of harmonics to average in the audio measurement.

The range is from 2 to 20.

Conditions

Measurement views: Audio Spectrum

Group

Sense commands

Syntax

```
[SENSe]:AUDio:AVERage:COUNt <average_value>
[SENSe]:AUDio:AVERage:COUNt?
```

Arguments

`<average_value> ::= <NR1>` the number of harmonics to average.

Examples

`SENSe:AUDIO:AVERAGE:COUNT 10` sets the number of harmonics to average to 10.

[SENSe]:AUDio:AVERage:ENABle

Enables or disables averaging of harmonics in the audio measurement.

Conditions

Measurement views: Audio Spectrum

Group

Sense commands

Syntax

```
[SENSe]:AUDio:AVERage:ENABle { OFF | ON | 0 | 1 }
[SENSe]:AUDio:AVERage:ENABle?
```

Arguments

`OFF` or `0` turns averaging off.

`ON` or `1` turns averaging on.

Examples

`SENSe:AUDIO:AVERAGE:ENABLE ON` turns on averaging of harmonics in the audio measurement.

[SENSe]:AUDio:BANDwidth|BWIDth

Sets or queries the audio measurement bandwidth.

Conditions

Measurement views: Audio Spectrum

Group

Sense commands

Syntax

```
[SENSe]:AUDio:BANDwidth|BWIDth <value>
[SENSe]:AUDio:BANDwidth|BWIDth?
```

Arguments

`<value> ::= <NR3>` the audio measurement bandwidth value.

Examples

`SENSe:AUDIO:BANDWIDTH|BWIDTH 20.5E+3` sets the audio measurement bandwidth to 20.5 kHz.

[SENSe]:AUDio:BANDwidth|BWIDth:RESolution

Sets or queries the resolution bandwidth for the audio measurement.

Conditions

Measurement views: Audio Spectrum

Group

Sense commands

Syntax

```
[SENSe]:AUDio:BANDwidth|BWIDth:RESolution <value>
[SENSe]:AUDio:BANDwidth|BWIDth:RESolution?
```

Arguments

`<value> ::= <NR3>` the resolution bandwidth value.

Examples

`SENSe:AUDIO:BANDWIDTH|BWIDTH:RESOLUTION 40` sets the resolution bandwidth to 40 Hz.

[SENSe]:AUDio:BANDwidth|BWIDth:RESolution:AUTO

Determines whether to set the resolution bandwidth frequency automatically or manually.

Conditions

Measurement views: Audio Spectrum

Group

Sense commands

Syntax

```
[SENSe]:AUDio:BANDwidth|BWIDth:RESolution:AUTO { OFF | ON | 0 | 1 }
[SENSe]:AUDio:BANDwidth|BWIDth:RESolution:AUTO?
```

Arguments

`OFF` or `0` specifies the resolution bandwidth is set manually.

`ON` or `1` specifies the resolution bandwidth is set automatically.

Examples

`SENSe:AUDIO:BANDWIDTH|BWIDTH:RESOLUTION:AUTO ON` sets the resolution bandwidth automatically.

[SENSe]:AUDio:BANDwidth|BWIDth:RESolution:SHAPe

Sets or queries the shape of the resolution bandwidth for the audio measurement.

Conditions

Measurement views: Audio Spectrum

Group

Sense commands

Syntax

```
[SENSe]:AUDio:BANDwidth|BWIDth:RESolution:SHAPe { FLATtop | KAISer }
[SENSe]:AUDio:BANDwidth|BWIDth:RESolution:SHAPe?
```

Arguments

`FLATtop` sets the resolution bandwidth shape to Flattop.

`KAISer` sets the resolution bandwidth shape to Kaiser.

Examples

`SENSe:AUDIO:BANDWIDTH|BWIDTH:RESOLUTION:SHAPE FLATtop` sets the resolution bandwidth to Flattop for the audio measurement.

[SENSe]:AUDio:CARRier:OFFSet

Sets or queries the carrier frequency offset. This selection is available for FM or PM measurements.

Conditions

Measurement views: Audio Spectrum

Group

Sense commands

Syntax

```
[SENSe]:AUDio:CARRier:OFFSet <value>
[SENSe]:AUDio:CARRier:OFFSet?
```

Arguments

`<value> ::= <NR3>` the carrier frequency offset value.

Returns

<NR3>

Examples

`SENSe:AUDIO:CARRIER:OFFSET 422.5` sets the carrier frequency offset to 422.5 Hz.

[SENSe]:AUDio:CARRier:OFFSet:AUTO

Determines whether to set the carrier frequency error automatically or manually. This selection is available FM or PM measurements.

Conditions

Measurement views: Audio Spectrum

Group

Sense commands

Syntax

```
[SENSe]:AUDio:CARRier:OFFSet:AUTO { OFF | ON | 0 | 1 }
[SENSe]:AUDio:CARRier:OFFSet:AUTO?
```

Arguments

OFF or 0 specifies the carrier frequency error is set manually.

ON or 1 specifies the carrier frequency error is set automatically.

Examples

SENSe:AUDIO:CARRIER:OFFSET:AUTO ON sets the carrier frequency error automatically.

[SENSe]:AUDio:DEMod:MUTE

Sets or queries the mute setting for the audio demodulation. Enabling Mute inhibits the audio from playing out through the PC speakers.

Conditions

Requires an RSA be connected with SignalVu-PC.

Group

Sense commands

Syntax

```
[SENSe]:AUDio:DEMod:MUTE { OFF | ON | 0 | 1 }
```

Arguments

ON or 1 enables the Mute setting for audio demodulation.

OFF or 0 disables the Mute setting for audio demodulation.

Returns

ON or 1 means that the Mute setting for audio demodulation is enabled.

OFF or 0 means that the Mute setting for audio demodulation is disabled.

Examples

SENSe:AUDIO:DEMOD:MUTE 1 enables the audio demodulation mute setting, inhibiting the audio signal from playing out through the PC speakers.

[SENSe]:AUDio:DEMod[:STATe]

Sets or queries the audio demodulation state.

Conditions

Requires an RSA be connected with SignalVu-PC.

Group

Sense commands

Syntax

```
[SENSe]:AUDio:DEMod[:STATe] { 0 | 1 }
```

Arguments

1 starts the demodulation.

0 stops the demodulation.

Returns

1 means the demodulation is started.

0 means the demodulation is stopped.

Examples

SENSe:AUDIO:DEMOD[:STATE] 1 starts the demodulating the audio signal.

[SENSe]:AUDio:DEMod:STReam:ACTivate

Sets or queries the Stream to File function. When checked, anytime the Run button is pressed, the audio stream-to-file begins.

Conditions

Requires an RSA be connected with SignalVu-PC.

Group

Sense commands

Syntax

```
[SENSe]:AUDio:DEMod:STReam:ACTivate { OFF | ON | 0 | 1 }
[SENSe]:AUDio:DEMod:STReam:ACTivate?
```

Arguments

ON or 1 enables the stream to file function.

OFF or 0 disables the stream to file function.

Returns

ON or 1 means the stream to file function is enabled.

OFF or 0 means the stream to file function is disabled.

Examples

SENSe:AUDIO:DEMOD:STREAM:ACTIVATE 1 enables (activates) the audio streaming function.

SENSe:AUDIO:DEMOD:STREAM:ACTIVATE? might return 0, indicating the audio streaming function is deactivated.

[SENSe]:AUDio:DEMod:STReam[:FILE] (No Query Form)

Sets the file name and file location of the audio file when streaming to disc.

Conditions

Requires an RSA be connected with SignalVu-PC.

Group

Sense commands

Syntax

```
[SENSe]:AUDio:DEMod:STReam[:FILE] <file>
```

Arguments

<file_name>::=<string> specifies the file name and location. The file extension is .wav. You can omit the extension.

You can use the absolute path to specify the file name. For example, specify the SAMPLE1 file in the My Documents folder on the C drive as "C:\My Documents\Audio1".

If you omit the directory path, the default path is used, which is C:\SignalVu-PC Files\Sample Data Records, initially.

Examples

SENSe:AUDIO:DEMOD:STREAM[:FILE] "C:\SignalVu-PC Files\Sample Data Records\Audio1" saves the streamed audio file Audio1 to the specified location.

[SENSe]:AUDio:DEMod:TYPE

Sets or queries both the audio signal demodulation type and bandwidth.

Conditions

Requires an RSA be connected with SignalVu-PC.

Group

Sense commands

Syntax

```
[SENSe]:AUDio:DEMod:TYPE { AM8K | FM8K | FM13K | FM75K | FM200K }
```

Related Commands

Arguments

AM8K

FM8K

FM13K

FM75K

FM200K

Returns

Same as arguments.

Examples

SENSe:AUDIO:DEMOD:TYPE AM8K sets the audio demodulation type to AM and the bandwidth to 8 KHz.

SENSe:AUDIO:DEMOD:TYPE? might return AM8K.

[SENSe]:AUDio:DEMod:VOLume

Sets or queries the volume of the demodulated audio signal when played out through the PC speakers.

Conditions

Requires an RSA be connected with SignalVu-PC.

Group

Sense commands

Syntax

```
[SENSe]:AUDio:DEMod:VOLume <value>
[SENSe]:AUDio:DEMod:VOLume?
```

Arguments

`<value>::=<NR2>` the value of relative volume setting.

Range = 0.0 to 1.0.

Examples

`SENSe:AUDIO:DEMOD:VOLUME 1.0` sets the volume output to maximum.

[SENSe]:AUDio:FILTer:DEEMphasis

Sets or queries the de-emphasis filter type.

Conditions

Measurement views: Audio Spectrum

Group

Sense commands

Syntax

```
[SENSe]:AUDio:FILTer:DEEMphasis { U25 | U50 | U75 | U750 | USER}
[SENSe]:AUDio:FILTer:DEEMphasis?
```

Arguments

`U25` specifies the 25 µs filter.

`U50` specifies the 50 µs filter.

`U75` specifies the 75 µs filter.

`U750` specifies the 750 µs filter.

`USER` specifies a user-defined filter.

Examples

`SENSe:AUDIO:FILTER:DEEMPHASIS U50` specifies the 50 µs filter for the audio measurement.

[SENSe]:AUDio:FILTer:DEEMphasis:ENABle

Enables or disables the de-emphasis filter for the audio measurement.

Conditions

Measurement views: Audio Spectrum

Group

Sense commands

Syntax

```
[SENSe]:AUDio:FILTer:DEEMphasis:ENABle { OFF | ON | 0 | 1 }
[SENSe]:AUDio:FILTer:DEEMphasis:ENABle?
```

Arguments

`OFF` or `0` turns off the de-emphasis filter.

`ON` or `1` turns on the de-emphasis filter.

Examples

`SENSe:AUDIO:FILTER:DEEMPHASIS:ENABLE ON` turns on the de-emphasis filter.

[SENSe]:AUDio:FILTer:DEEMphasis:USER

Sets or queries a user-defined de-emphasis audio filter.

Conditions

Measurement views: Audio Spectrum

Group

Sense commands

Syntax

```
[SENSe]:AUDio:FILTer:DEEMphasis:USER <value>
[SENSe]:AUDio:FILTer:DEEMphasis:USER?
```

Arguments

`<value> ::= <NR3>` the value of the user defined filter.

Examples

`SENSe:AUDIO:FILTER:DEEMPHASIS:USER 50.0E-6` sets the user-defined de-emphasis filter to 50 µs.

[SENSe]:AUDio:FILTer:HPF

Sets or queries the high-pass filter type.

Conditions

Measurement views: Audio Spectrum

Group

Sense commands

Syntax

```
[SENSe]:AUDio:FILTer:HPF { H20 | H50 | H300 | H400 | USER }
[SENSe]:AUDio:FILTer:HPF?
```

Arguments

`H20` specifies the 20 Hz filter.

`H50` specifies the 50 Hz filter.

`H300` specifies the 300 Hz filter.

`H400` specifies the 400 Hz filter.

`USER` specifies a user-defined filter.

Examples

`SENSe:AUDIO:FILTER:HPF H300` specifies the 300 Hz filter for the audio measurement.

[SENSe]:AUDio:FILTer:HPF:ENABle

Enables or disables the high-pass audio filter for the audio measurement.

Conditions

Measurement views: Audio Spectrum

Group

Sense commands

Syntax

```
[SENSe]:AUDio:FILTer:HPF:ENABle { OFF | ON | 0 | 1 }
[SENSe]:AUDio:FILTer:HPF:ENABle?
```

Arguments

`OFF` or `0` turns the high-pass audio filter off.

`ON` or `1` turns the high-pass audio filter on.

Examples

`SENSe:AUDIO:FILTER:HPF:ENABLE ON` turns the high-pass audio filter on.

[SENSe]:AUDio:FILTer:HPF:USER

Sets or queries a user-defined high-pass audio filter.

Conditions

Measurement views: Audio Spectrum

Group

Sense commands

Syntax

```
[SENSe]:AUDio:FILTer:HPF:USER <value>
[SENSe]:AUDio:FILTer:HPF:USER?
```

Arguments

`<value> ::= <NR3>` the value of the user defined filter.

Examples

`SENSe:AUDIO:FILTER:HPF:USER 50` sets the user-defined high-pass filter to 50 Hz.

[SENSe]:AUDio:FILTer:LPF

Sets or queries the low-pass filter type.

Conditions

Measurement views: Audio Spectrum

Group

Sense commands

Syntax

```
[SENSe]:AUDio:FILTer:LPF { H300 | H3K | H15K | H30K | H80K | H300K | USER }
[SENSe]:AUDio:FILTer:LPF?
```

Arguments

`H300` specifies the 300 Hz filter.

`H3K` specifies the 3 kHz filter.

`H15` specifies the 15 kHz filter.

`H30K` specifies the 30 kHz filter.

`H80K` specifies the 80 kHz filter.

`H300K` specifies the 300 kHz filter.

`USER` specifies a user-defined filter.

Examples

`SENSe:AUDIO:FILTER:LPF H300` specifies the 300 Hz filter for the audio measurement.

[SENSe]:AUDio:FILTer:LPF:ENABle

Enables or disables the low-pass filter for the audio measurement.

Conditions

Measurement views: Audio Spectrum

Group

Sense commands

Syntax

```
[SENSe]:AUDio:FILTer:LPF:ENABle { OFF | ON | 0 | 1 }
[SENSe]:AUDio:FILTer:LPF:ENABle?
```

Arguments

`OFF` or `0` turns the low-pass audio filter off.

`ON` or `1` turns the low-pass audio filter on.

Examples

`SENSe:AUDIO:FILTER:LPF:ENABLE ON` turns the low-pass filter on.

[SENSe]:AUDio:FILTer:LPF:USER

Sets or queries a user-defined low-pass audio filter.

Conditions

Measurement views: Audio Spectrum

Group

Sense commands

Syntax

```
[SENSe]:AUDio:FILTer:LPF:USER <value>
[SENSe]:AUDio:FILTer:LPF:USER?
```

Arguments

`<value> ::= <NR3>` the value of the user defined filter.

Examples

`SENSe:AUDIO:FILTER:LPF:USER 1500` sets the user-defined low-pass filter to 1.50 kHz.

[SENSe]:AUDio:FILTer:MODE:PREDefined

Selects a predefined audio filter or a filter stored on the instrument file system.

Conditions

Measurement views: Audio Spectrum

Group

Sense commands

Syntax

```
[SENSe]:AUDio:FILTer:MODE:PREDefined { OFF | ON | 0 | 1 }
[SENSe]:AUDio:FILTer:MODE:PREDefined?
```

Arguments

`OFF` or `0` specifies the predefined filter on the file system. The file is specified by the `[SENSe]:AUDio:FILTer:SOURce?` command.

`ON` or `1` specifies the filter will be one of the selected predefined filters.

Examples

`SENSe:AUDIO:FILTER:MODE:PREDEFINED ON` specifies the filter will be one of the predefined filters.

[SENSe]:AUDio:FILTer:SOURce? (Query Only)

Queries the measurement filter from the user-defined source file.

Conditions

Measurement views: Audio Spectrum

Group

Sense commands

Syntax

```
[SENSe]:AUDio:FILTer:SOURce?
```

Related Commands

[SENSe]:AUDio:FILTer:MODE:PREDefined (on page 1971)

Arguments

None

Returns

`<file name> ::= <string>` the name of the predefined filter.

Examples

`SENSe:AUDIO:FILTER:SOURCE? AUD-Test3` loads the predefined filter “AUD-Test3.”

[SENSe]:AUDio:FILTer:STANdard

Sets or queries the Standard audio filter.

Conditions

Measurement views: Audio Spectrum

Group

Sense commands

Syntax

```
[SENSe]:AUDio:FILTer:STANdard { CCITT | CMSG }
[SENSe]:AUDio:FILTer:STANdard?
```

Arguments

`CCITT` specifies the CCITT Standard filter.

`CMSG` specifies the CMSG Standard filter.

Examples

`SENSe:AUDIO:FILTER:STANDARD CCITT` specifies the CCITT Standard filter.

[SENSe]:AUDio:FILTer:STANdard:ENABle

Enables or disables a Standard filter for the audio measurement.

Conditions

Measurement views: Audio Spectrum

Group

Sense commands

Syntax

```
[SENSe]:AUDio:FILTer:STANdard:ENABle { OFF | ON | 0 | 1 }
[SENSe]:AUDio:FILTer:STANdard:ENABle?
```

Arguments

`OFF` or `0` turns the standard filter off.

`ON` or `1` turns the standard audio filter on.

Examples

`SENSe:AUDIO:FILTER:STANDARD:ENABLE ON` turns the Standard filter on.

[SENSe]:AUDio:FREQuency

Sets or queries the reference audio frequency.

Conditions

Measurement views: Audio Spectrum

Group

Sense commands

Syntax

```
[SENSe]:AUDio:FREQuency <value>
[SENSe]:AUDio:FREQuency?
```

Arguments

`<value> ::= <NR3>` the reference audio frequency.

Examples

`SENSe:AUDIO:FREQUENCY?` might return `5.8240000000E+3` indicating that the reference audio frequency is 5.824 kHz.

[SENSe]:AUDio:FREQuency:AUTO

Determines whether to set the reference audio frequency automatically or manually.

Conditions

Measurement views: Audio Spectrum

Group

Sense commands

Syntax

```
[SENSe]:AUDio:FREQuency:AUTO { OFF | ON | 0 | 1 }
[SENSe]:AUDio:FREQuency:AUTO?
```

Arguments

`OFF` or `0` specifies the audio reference frequency is set manually.

`ON` or `1` specifies the audio reference frequency is set automatically.

Examples

`SENSe:AUDIO:FREQUENCY:AUTO ON` sets the audio reference frequency automatically.

[SENSe]:AUDio:HNOise:ENABle

Enables or disables the Hum and Noise information in the Audio Summary display.

Conditions

Measurement views: Audio Summary

Group

Sense commands

Syntax

```
[SENSe]:AUDio:HNOise:ENABle { OFF | ON | 0 | 1 }
[SENSe]:AUDio:HNOise:ENABle?
```

Related Commands

[SENSe]:AUDio:HNOise:REFerence (on page 1974)

Arguments

`OFF` or `0` turns off the Hum and Noise information in Audio Summary display.

`ON` or `1` turns on the Hum and Noise information in the Audio Summary display.

Examples

`SENSe:AUDIO:HNOISE:ENABLE OFF` turns off the Hum and Noise information in the Audio Spectrum display.

[SENSe]:AUDio:HNOise:REFerence (No Query Form)

Captures the current Hum and Noise information in the Audio Summary display.

Conditions

Measurement views: Audio Summary

Group

Sense commands

Syntax

```
[SENSe]:AUDio:HNOise:REFerence
```

Related Commands

[SENSe]:AUDio:HNOise:ENABle (on page 1973)

Arguments

None

Examples

`SENSe:AUDIO:HNOISE:REFERENCE` shows the current Hum and Noise information in the Audio Summary display.

[SENSe]:AUDio:SIGnal:TYPE

Sets or queries the audio signal type.

The signal type selection determines the available selections for the `[SENSe]:AUDio:UNITs` command.

Conditions

Measurement views: Audio Spectrum

Group

Sense commands

Syntax

```
[SENSe]:AUDio:SIGnal:TYPE { AM | FM | DIRect | PM }
[SENSe]:AUDio:SIGnal:TYPE?
```

Related Commands

[SENSe]:AUDio:UNITs (on page 1976)

Arguments

`AM` sets the signal type to amplitude modulation.

`FM` sets the signal type to frequency modulation.

`DIRect` sets the signal type to direct modulation.

`PM` sets the signal type to pulse modulation.

Returns

Examples

`SENSe:AUDIO:SIGNAL:TYPE FM` sets the signal type to FM, frequency modulation for the audio spectrum measurement.

[SENSe]:AUDio:SPECtrum:CLEar:RESults (No Query Form)

Clears the results of the audio spectrum measurement.

Conditions

Measurement views: Audio Spectrum

Group

Sense commands

Syntax

```
[SENSe]:AUDio:SPECtrum:CLEar:RESults
```

Arguments

None

[SENSe]:AUDio:SPECtrum:POINts:COUNt

Sets or queries the number of trace points acquired for the audio spectrum display.

Conditions

Measurement views: Audio Spectrum

Group

Sense commands

Syntax

```
[SENSe]:AUDio:SPECtrum:POINts:COUNt { P801 | P2401 | P4001 | P8001 | P10401 }
[SENSe]:AUDio:SPECtrum:POINts:COUNt?
```

Arguments

`P801` sets the number of sample points to 801.

`P2401` sets the number of sample points to 2401.

`P4001` sets the number of sample points to 4001.

`P8001` sets the number of sample points to 8001.

`P10401` sets the number of sample points to 10401.

Examples

`SENSe:AUDIO:SPECTRUM:POINTS:COUNT P2401` sets the number of sample points to 2401 in the audio spectrum display.

[SENSe]:AUDio:UNITs

Sets or queries the units of the audio spectrum measurement.

The available selections depend on signal type.

Conditions

Measurement views: Audio Spectrum

Group

Sense commands

Syntax

```
[SENSe]:AUDio:UNITs { AM | AM2 | DBAM2 | HZ | HZ2 | DBHZ2 | RAD | RAD2 | DBRAD2 | DBM | DBV | VOLTs
    | WATTs }
[SENSe]:AUDio:UNITs?
```

Related Commands

[SENSe]:AUDio:SIGnal:TYPE (on page 1974)

Arguments

The following table lists the arguments.

Argument	Unit	Signal type
DBM	dBm	Direct
DBV	dBV	
VOLTs	Volts	
WATTs	Watts	
AM	am	AM
AM2	am^2	
DBAM2	$dBam^2$	

Argument	Unit	Signal type
HZ	Hz	FM
HZ2	Hz[22]	
DBHZ2	dBHz[2]	
RAD	rad	PM
RAD2	rad[2]	
DBRAD2	dBrad[2]	

Examples

`SENSe:AUDIO:UNITS VOLTs` sets the units to volts for the Direct signal type.

[SENSe]:AVTime[:MEASview<y>]:SELEcted:SOURce

Sets or queries the Source of Amplitude versus Time for a selected Measurement view number.

Conditions

Measurement views: Amplitude versus Time

Source cannot be changed when acquisition or analysis is in progress. Stop the acquisition and change the Source.

Group

Sense commands

Syntax

```
[SENSe]:AVTime[:MEASview<y>]:SELEcted:SOURce {"Source 1" | "Source 2" | "Source 3" | "Source 4"}
[SENSe]:AVTime[:MEASview<y>]:SELEcted:SOURce?
```

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`SENSe:AVTIME:MEASVIEW1:SELECTED:SOURCE 2` queries the Source for Amplitude versus Time Measurement View 1.

[SENSe]:AVTime[:MEASview<y>]:SPAN:LINK

Sets or queries the frequency span in the Amplitude versus Time measurement. Programming a specified span disables the `[SENSe]:AVTime[:MEASview<y>]:{BANDwidth|BWIDth}` setting.

Conditions

Measurement views: Amplitude versus Time

Group

Sense commands

Syntax

```
[SENSe]:AVTime[:MEASview<y>]:SPAN:LINK <value>
[SENSe]:AVTime[:MEASview<y>]:SPAN:LINK?
```

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

OFF or 0 specifies that Link to Span is not checked

ON or 1 specifies that Link to Span is checked

Examples

SENSe:AVTIME:MEASVIEW1:SPAN:LINK? the frequency span link on Amplitude versus Time Measurement View 1.

[SENSe]:AVTime[:MEASview<y>]:{BANDwidth|BWIDth}

Sets or queries the time-domain bandwidth filter in the Amplitude versus Time measurement. Programming a specified bandwidth disables the [SENSe]:AVTime[:MEASview<y>]:SPAN setting.

Conditions

Measurement views: Amplitude versus Time

Group

Sense commands

Syntax

```
[SENSe]:AVTime[:MEASview<y>]:{BANDwidth|BWIDth} <value>
[SENSe]:AVTime[:MEASview<y>]:{BANDwidth|BWIDth}?
```

Arguments

<value>::<NRf> specifies the filter bandwidth.

Range: 1 Hz to 20 MHz (Standard) / 60 MHz (Option 110).

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

SENSe:AVTIME:MEASVIEW1:BANDWIDTH 10MHz sets the filter bandwidth to 10 MHz on Amplitude versus Time Measurement View 1.

[SENSe]:AVTime[:MEASview<y>]:{BANDwidth|BWIDth}:ACTual? (Query Only)

Queries the actual time-domain bandwidth in the Amplitude versus Time measurement.

Conditions

Measurement views: Amplitude versus Time

Group

Sense commands

Syntax

```
[SENSe]:AVTime[:MEASview<y>]:{BANDwidth|BWIDth}:ACTual?
```

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

<NRf> The actual time-domain bandwidth in Hz.

Examples

SENSe:AVTIME:MEASVIEW1:BANDWIDTH:ACTUAL? might return 20E+6, indicating that the actual time-domain bandwidth is 20 MHz on Amplitude versus Time Measurement View 1.

[SENSe]:AVTime[:MEASview<y>]:CLEar:RESults (No Query Form)

Restarts multi-trace functions (Average and Max/Min Hold).

Conditions

Measurement views: Amplitude versus Time

Group

Sense commands

Syntax

```
[SENSe]:AVTime[:MEASview<y>]:CLEar:RESults
```

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

SENSe:AVTIME:MEASVIEW1:CLEAR:RESULTS restarts multi-trace functions on Amplitude versus Time Measurement View 1.

[SENSe]:AVTime[:MEASview<y>]:COLor

Sets or queries the color palette of three-dimensional graphs.

Conditions

Measurement views: Amplitude versus Time

Group

Sense commands

Syntax

```
[SENSe]:AVTime[:MEASview<y>]:COLor {RED | GREen | BLUe | CYAN | BCYan | YELLow | MAGenta
| GRAY | TEMPerature | SPECtral}
```

```
[SENSe]:AVTime[:MEASview<y>]:COLor?
```

Arguments

The following table lists the arguments:

Argument	Palette
RED	Red
GREen	Green
BLUe	Blue
CYAN	Cyan
BCYan	Binary cyan
YELLow	Yellow
MAGenta	Magenta
GRAY	Gray
TEMPerature	Temperature
SPECtral	Spectral

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`SENSE:AVTIME:COLOR TEMPERATURE` selects the temperature color palette.

[SENSe]:AVTime[:MEASview<y>]:COLor:MAXimum

Sets or queries the maximum value of the color axis in the Amplitude versus Time measurement.

Conditions

Measurement views: Amplitude versus Time

Group

Sense commands

Syntax

`[SENSe]:AVTime[:MEASview<y>]:COLor:MAXimum <value>`

`[SENSe]:AVTime[:MEASview<y>]:COLor:MAXimum?`

Arguments

`<value>::=<NRf>` specifies the maximum value of the color axis.

Range: The minimum value to 100%.

The minimum value is set using the `[SENSe]:AVTime[:MEASview<y>]:COLor:MINimum` command.

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`SENSE:AVTIME:COLOR:MAXIMUM 90` sets the maximum value of the color axis to 90%.

[SENSe]:AVTime[:MEASview<y>]:COLor:MINimum

Sets or queries the minimum value of the color axis in the Amplitude versus Time measurement.

Conditions

Measurement views: Amplitude versus Time

Group

Sense commands

Syntax

`[SENSe]:AVTime[:MEASview<y>]:COLor:MINimum <value>`

`[SENSe]:AVTime[:MEASview<y>]:COLor:MINimum?`

Arguments

`<value>::=<NRf>` specifies the minimum value of the color axis.

Range: 0% to the maximum value.

The maximum value is set using the `[SENSe]:AVTime[:MEASview<y>]:COLor:MAXimum` command.

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`SENSE:AVTIME:COLOR:MINIMUM 10` sets the minimum value of the color axis to 10%.

[SENSe]:AVTime[:MEASview<y>]:COLor:REVerse[:STATe]

Determines whether to reverse the color palette in the Amplitude versus Time view. When reversed, the highest density takes the color normally used for the lowest.

Conditions

Measurement views: Amplitude versus Time

Group

Sense commands

Syntax

`[SENSe]:AVTime[:MEASview<y>]:COLor:REVerse[:STATe] {OFF | ON | 0 | 1}`

`[SENSe]:AVTime[:MEASview<y>]:COLor:REVerse[:STATe]?`

Arguments

`OFF` or `0` uses the color palette in its normal order.

ON or 1 reverses the color palette.

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

SENSE:AVTIME:COLOR:REVERSE:STATE ON reverses the color palette.

[SENSe]:AVTime[:MEASview<y>]:MAXTracepoints

Sets or queries the maximum trace points in the Amplitude versus Time measurement.

Conditions

Measurement views: Amplitude versus Time

Group

Sense commands

Syntax

```
[SENSe]:AVTime[:MEASview<y>]:MAXTracepoints { ONEK | TENK | HUNDredk | NEVerdecimate }
[SENSe]:AVTime[:MEASview<y>]:MAXTracepoints?
```

Arguments

ONEK sets the maximum trace points to 1 k.

TENK sets the maximum trace points to 10 k.

HUNDredk sets the maximum trace points to 100 k.

NEVerdecimate never decimates the trace points.

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

SENSe:AVTIME:MEASVIEW1:MAXTRACEPOINTS TENK sets the maximum trace points to 10 k on Amplitude versus Time Measurement View 1.

[SENSe]:AVTime[:MEASview<y>]:METHod

Sets or queries the method to set the measurement bandwidth in the Amplitude versus Time measurement.

Conditions

Measurement views: Amplitude versus Time

Group

Sense commands

Syntax

```
[SENSe]:AVTime[:MEASview<y>]:METHod { SPAN | TDBW }
[SENSe]:AVTime[:MEASview<y>]:METHod?
```

Arguments

SPAN specifies that the measurement bandwidth is set by the frequency span, using the [SENSe]:AVTime:SPAN command.

TDBW specifies that the measurement bandwidth is set by the time-domain bandwidth, using the [SENSe]:AVTime[:MEASview<y>]:{BANDwidth|BWIDth} command.

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

SENSe:AVTIME:MEASVIEW1:METHOD SPAN specifies that the measurement bandwidth is set by the frequency span on Amplitude versus Time Measurement View 1.

[SENSe]:AVTime[:MEASview<y>]:OGRAm:COLor

Sets or queries the ogram color scheme.

Conditions

Applicable measurement view must be active.

Group

Sense commands

Syntax

[SENSe]:AVTime[:MEASview<y>]:OGRAm:COLor {RED | GREen | BLUe | CYAN | BCYan | YELLow | MAGenta | GRAY | TEMPerature | SPECtral}

[SENSe]:AVTime[:MEASview<y>]:OGRAm:COLor?

Related Commands

[SENSe]:AVTime[:MEASview<y>]:OGRAm:COLor:MAXimum (on page 1984)

Arguments

The following table lists the arguments:

Argument	Palette
RED	Red
GREen	Green
BLUe	Blue
CYAN	Cyan
BCYan	Binary cyan
YELLow	Yellow
MAGenta	Magenta
GRAY	Gray

Argument	Palette
TEMPerature	Temperature
SPECtral	Spectral

Parameter <y> = 1 to 8 represents measurement view index. MEASview is optional and defaults to 1.

Returns

{RED | GREen | BLUe | CYAN | BCYan | YELLow | MAGenta | GRAY | TEMPerature | SPECtral}

Examples

SENSe:AVTime:MEASview1:OGRAm:COLor TEMPerature selects the temperature color palette.

[SENSe]:AVTime[:MEASview<y>]:OGRAm:COLor:MAXimum

Sets or queries the maximum value of the color axis in the ogram.

Conditions

Applicable measurement view must be active.

Group

Sense commands

Syntax

[SENSe]:AVTime[:MEASview<y>]:OGRAm:COLor:MAXimum <value>

[SENSe]:AVTime[:MEASview<y>]:OGRAm:COLor:MAXimum?

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional. By default y is 1.

<value>::=<NRf> specifies the maximum value of the color axis.

Range: -100 to +100 dBm.

Related Commands

[SENSe]:AVTime[:MEASview<y>]:OGRAm:COLor:MINimum (on page 1984)

Returns

<NRf> specifies the maximum value of the color axis.

Examples

SENSe:AVTime:MEASview1:OGRAm:COLor:MAXimum 10 sets the maximum value of the color axis to 10 dBm.

[SENSe]:AVTime[:MEASview<y>]:OGRAm:COLor:MINimum

Sets or queries the minimum value of the color axis in the ogram.

Conditions

Applicable measurement view must be active.

Group

Sense commands

Syntax

`[SENSe]:AVTime[:MEASview<y>]:OGRAm:COLor:MINimum <value>`

`[SENSe]:AVTime[:MEASview<y>]:OGRAm:COLor:MINimum?`

Related Commands

[SENSe]:AVTime[:MEASview<y>]:OGRAm:COLor:MAXimum (on page 1984)

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional. By default y is 1.

`<value>::=` `<NRf>` specifies the minimum value of the color axis.

Range: -100 to +100 dBm.

Returns

<NRf> specifies the minimum value of the color axis.

Examples

`SENSe:AVTime:MEASview1:OGRAm:COLor:MINimum 10` sets the minimum value of the color axis to 10 dBm.

[SENSe]:AVTime[:MEASview<y>]:OGRAm:COLor:REVerse

Sets or queries reverse color scheme state.

Conditions

Applicable measurement view must be active.

Group

Sense commands

Syntax

`[SENSe]:AVTime[:MEASview<y>]:OGRAm:COLor:REVerse {OFF | ON | 0 | 1}`

`[SENSe]:AVTime[:MEASview<y>]:OGRAm:COLor:REVerse?`

Related Commands

[SENSe]:AVTime[:MEASview<y>]:OGRAm:COLor (on page 1983)

Arguments

`OFF` or `0` disables Reverse color scheme.

`ON` or `1` enables Reverse color scheme.

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional. By default y is 1.

Returns

`0` disables Reverse color scheme.

`1` enables Reverse color scheme.

Examples

`SENSe:AVTime:MEASview1:OGRAm:COLor:REVerse OFF` disables reverse color scheme.

[SENSe]:AVTime[:MEASview<y>]:PLOT

Sets or queries the Amplitude vs time display mode.

Conditions

Applicable measurement view must be active.

Group

Sense commands

Syntax

`[SENSe]:AVTime[:MEASview<y>]:PLOT {TRACE | OGRAM | SPLIT}`

`[SENSe]:AVTime[:MEASview<y>]:PLOT?`

Arguments

`TRACE` displays time plot .

`OGRAM` displays Ogram plot.

`SPLIT` displays both ogram and time plots.

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Returns

`{TRACE | OGRAM | SPLIT}`

Examples

`SENSe:AVTime:MEASview1:PLOT TRACE` sets plot type to TRACE (Time plot).

[SENSe]:AVTime[:MEASview<y>]:SPAN

Sets or queries the frequency span in the Amplitude versus Time measurement. Programming a specified span disables the `[SENSe]:AVTime[:MEASview<y>]:{BANDwidth|BWIDth}` setting.

Conditions

Measurement views: Amplitude versus Time

Group

Sense commands

Syntax

`[SENSe]:AVTime[:MEASview<y>]:SPAN <value>`
`[SENSe]:AVTime[:MEASview<y>]:SPAN?`

Arguments

`<value>::<NRf>` specifies the frequency span.

Range: 10 Hz to 40 MHz (Standard) / 110 MHz (Option 110)

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`SENSe:AVTIME:MEASVIEW1:SPAN 5MHz` sets the frequency span to 5 MHz on Amplitude versus Time Measurement View 1.

[SENSe]:BLUEtooth:ANALysis:LENGth

Sets or returns the analysis length in the Bluetooth displays. Programming a specified length sets `[SENSe]:BLUEtooth:ANALysis:LENGth:AUTO` to off.

Conditions

Measurement views: All Bluetooth displays

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:ANALysis:LENGth <value>
[SENSe]:BLUEtooth:ANALysis:LENGth?
```

Related Commands

[SENSe]:BLUEtooth:ANALysis:LENGth:AUTO (on page 1988)

Arguments

`<value> ::= <NRf>` the analysis length. The minimum range depends on the modulation type.

Examples

`SENSe:BLUETOOTH:ANALYSIS:LENGTH 25.625E-6` sets the analysis length to 25.625 µs.

[SENSe]:BLUEtooth:ANALysis:LENGth:ACTual? (Query Only)

Sets the actual analysis length in the Bluetooth displays.

Conditions

Measurement views: All Bluetooth displays

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:ANALysis:LENGth:ACTual?
```

Arguments

None

Returns

<NR3> the actual analysis length in seconds.

Examples

SENSe:BLUETOOTH:ANALYSIS:LENGTH:ACTUAL? might return 25.625E-6, indicating that the actual analysis length is 25.625 µs.

[SENSe]:BLUEtooth:ANALysis:LENGth:AUTO

Sets or queries the length mode of the Bluetooth displays.

Conditions

Measurement views: All Bluetooth displays

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:ANALysis:LENGth:AUTO { OFF | ON | 0 | 1 }
[SENSe]:BLUEtooth:ANALysis:LENGth:AUTO?
```

Related Commands

[SENSe]:BLUEtooth:ANALysis:LENGth (on page 1987)

Arguments

OFF or 0 sets the analysis length manually, using the [SENSe]:BLUEtooth:ANALysis:LENGth command.

ON or 1 sets the analysis length automatically.

Examples

SENSe:BLUETOOTH:ANALYSIS:LENGTH:AUTO ON sets the analysis length mode automatically.

[SENSe]:BLUEtooth:ANALysis:MEAS[:BANDwidth|BWIDth]

Sets or returns the measurement bandwidth in the Bluetooth measurement.

Conditions

Measurement views: All Bluetooth displays

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:ANALysis:MEAS[:BANDwidth|BWIDth] <value>
[SENSe]:BLUEtooth:ANALysis:MEAS[:BANDwidth|BWIDth]?
```

Arguments

<value> ::= <NR3>, the measurement bandwidth.

Examples

SENSe:BLUETOOTH:ANALYSIS:MEAS[:BANDWIDTH|BWIDTH]? might return 6.400E+6 indicating the measurement bandwidth is 6.400 MHz.

[SENSe]:BLUEtooth:ANALysis:MEAS:SOURce

Sets or returns the analysis measurement source in the Bluetooth measurement.

Conditions

Measurement views: All Bluetooth displays

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:ANALysis:MEAS:SOURce { MANual | AUTO | LSP }
[SENSe]:BLUEtooth:ANALysis:MEAS:SOURce?
```

Arguments

MANual allows for manual selection of the measurement bandwidth. A typical use is setting a narrow measurement bandwidth on a small range of frequencies to examine one signal while rejecting the others that are present in a Spectrum graph.

AUTO automatically selects a measurement bandwidth. The measurement picks a bandwidth based on other parameter settings (such as symbol rate and modulation type).

LSP sets the measurement bandwidth to Link to Span. This allows you to use the Spectrum display to tune the frequency, view the signal, and then use the Span control to set the measurement bandwidth. This mode emulates legacy instruments.

Examples

SENSe:BLUETOOTH:ANALYSIS:MEAS:SOURCE? might return MAN indicating a manual selection of the bandwidth measurement.

[SENSe]:BLUEtooth:ANALysis:OFFSet

Sets or returns the Bluetooth symbol analysis offset.

Conditions

Measurement views: Bluetooth displays

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:ANALysis:OFFSet <value>
[SENSe]:BLUEtooth:ANALysis:OFFSet?
```

Arguments

<NR1> the symbol analysis offset as a percentage. The range is from 0 to 100%.

Examples

SENSe:BLUETOOTH:ANALYSIS:OFFSET -50 sets the symbol analysis of set to -50%.

[SENSe]:BLUEtooth:ANALysis:OFFSet:AUTO

Determines whether to set the carrier frequency error automatically or manually in the Bluetooth displays.

Conditions

Measurement views: Bluetooth displays

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:ANALysis:OFFSet:AUTO { OFF | ON | 0 | 1 }
[SENSe]:BLUEtooth:ANALysis:OFFSet:AUTO?
```

Arguments

`OFF` or `0` specifies the carrier frequency error is set manually.

`ON` or `1` specifies the carrier frequency error is set automatically.

Examples

`SENSe:BLUETOOTH:ANALYSIS:OFFSET:AUTO ON` sets the carrier frequency error automatically.

[SENSe]:BLUEtooth:ANALysis:TIMe:UNITs (No Query Form)

Sets the time units for the Bluetooth Constellation settings.

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:ANALysis:TIMe:UNITs {SEConds | SYMBols }
[SENSe]:BLUEtooth:ANALysis:TIMe:UNITs?
```

Arguments

`SEConds` sets the time units to seconds.

`SYMBols` sets the time units to symbols.

Examples

`SENSe:BLUETOOTH:ANALYSIS:TIME:UNITS` sets the time units to seconds.

[SENSe]:BLUEtooth([:BANDwidth]|:BWIDth):MODE

Sets or returns the measurement bandwidth mode for the Bluetooth measurements.

Conditions

Measurement views: Any Bluetooth display

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth([:BANDwidth]|:BWIDth):MODE { MANual | AUTO |LSP }
[SENSe]:BLUEtooth([:BANDwidth]|:BWIDth):MODE?
```

Arguments

`MANual` allows for manual selection of the measurement bandwidth. A typical use is setting a narrow measurement bandwidth on a small range of frequencies to examine one signal while rejecting the others that are present in a Spectrum graph.

`AUTO` automatically selects a measurement bandwidth. The measurement picks a bandwidth based on other parameter settings (such as symbol rate and modulation type).

`LSP` sets the Measurement BW to Link to Span. This allows you to use the Spectrum display to tune the frequency, view the signal, and then use the Span control to set the measurement bandwidth. This mode emulates legacy instruments.

Examples

`SENSe:BLUETOOTH([:BANDWIDTH]|:BWIDTH):MODE AUTO` sets the bandwidth mode to automatically select a measurement bandwidth.

[SENSe]:BLUEtooth([:BANDwidth]|:BWIDth):TINTerval

Sets or returns the measurement bandwidth frequency span in the Bluetooth displays.

Conditions

Measurement views: Any Bluetooth display

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth([:BANDwidth]|:BWIDth):TINTerval <value>
[SENSe]:BLUEtooth([:BANDwidth]|:BWIDth):TINTerval?
```

Arguments

`<value> ::= <NRf>` specifies the measurement bandwidth in MHz.

Returns

`<NRf>` the actual measurement bandwidth.

Examples

`SENSe:BLUETOOTH([:BANDWIDTH]|:BWIDTH):TINTERVAL?` might return `35.255E+6` indicating the bandwidth is 35.255 MHz.

[SENSe]:BLUEtooth([:BANDwidth]|:BWIDth):TINTerval:AUTO

Sets or returns the Bluetooth measurement bandwidth to Auto or to Manual.

Conditions

Measurement views: Any Bluetooth display

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth([:BANDwidth]|:BWIDth):TINTerval:AUTO { 0 | 1 | OFF | ON }
[SENSe]:BLUEtooth([:BANDwidth]|:BWIDth):TINTerval:AUTO?
```

Arguments

`OFF` or `0` sets measurement bandwidth to Auto.

`ON` or `1` sets measurement bandwidth to Manual.

Examples

`SENSe:BLUETOOTH([:BANDWIDTH]|:BWIDTH):TINTERVAL:AUTO?` might return `1` indicating the measurement bandwidth is set to Auto.

[SENSe]:BLUEtooth:CONStellation:PREFs:GRATicule:SHOW

Show or hides the graticule state in the Bluetooth Constellation display.

Conditions

Measurement views: Bluetooth Constellation

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:CONStellation:PREFs:GRATicule:SHOW { OFF | ON | 0 | 1 }
[SENSe]:BLUEtooth:CONStellation:PREFs:GRATicule:SHOW?
```

Arguments

`OFF` or `0` hides the graticule in the display.

`ON` or `1` shows the graticule in the display.

Returns

`0` indicates the graticule is turned off.

`1` indicates the graticule is turned on.

Examples

`SENSe:BLUETOOTH:CONSTELLATION:PREFS:GRATICULE:SHOW ON` turns on the graticule in the display.

[SENSe]:BLUEtooth:CONStellation:PREFs:MARKERS:SHOW

Show or hides the marker readout in the Bluetooth Constellation display.

Conditions

Measurement views: Bluetooth Constellation

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:CONStellation:PREFs:MARKERS:SHOW { OFF | ON | 0 | 1 }
[SENSe]:BLUEtooth:CONStellation:PREFs:MARKERS:SHOW?
```

Arguments

`OFF` or `0` hides the marker readout in the display.

`ON` or `1` shows the marker readout in the display.

Returns

`0` indicates the marker readout is turned off.

`1` indicates the marker readout is turned on.

Examples

`SENSe:BLUETOOTH:CONSTELLATION:PREFS:GRATICULE:SHOW ON` turns on the marker readout in the display.

[SENSe]:BLUEtooth:CONStellation:PREFs:RADIx

Sets or returns the radix of the marker readout in the Bluetooth Constellation display.

Conditions

Measurement views: Bluetooth Constellation

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:CONStellation:PREFs:RADIx { Binary | Hex }
[SENSe]:BLUEtooth:CONStellation:PREFs:RADIx?
```

Arguments

`Binary` sets the marker readout to Binary.

`Hex` sets the marker readout to Hex.

Examples

`SENSe:BLUETOOTH:CONSTELLATION:PREFS:RADIX?` might return BINARY indicating the Marker readout radix is set to Binary.

[SENSe]:BLUEtooth:CONStellation:TRACE<x>:CONTent

Sets or returns the appearance of the specified trace (Trace 1 or Trace 2) in the Bluetooth Constellation display.

Conditions

Measurement views: Bluetooth Constellation

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:CONStellation:TRACE<x>:CONTent { Vectors | Points | Lines }
[SENSe]:BLUEtooth:CONStellation:TRACE<x>:CONTent?
```

Arguments

`Vectors` indicates the traces appear as vectors.

`Points` indicates the traces appear as a series of points.

`Lines` indicates the traces appear as lines.

Examples

`SENSe:BLUETOOTH:CONSTELLATION:TRACE1:CONTENT VECTORS` sets the Trace 1 to show as vectors in the display.

[SENSe]:BLUEtooth:CONStellation:TRACe<x>:FREEze

Sets or freezes the updates of the specified trace (Trace 1 or Trace 2) in the Bluetooth Constellation display.

Conditions

Measurement views: Bluetooth Constellation

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:CONStellation:TRACe<x>:FREEze { OFF | ON | 0 | 1 }
[SENSe]:BLUEtooth:CONStellation:TRACe<x>:FREEze?
```

Arguments

`OFF` or `0` normally updates the display of the specified trace.

`ON` or `1` stops updating the display of the specified trace.

Examples

`SENSe:BLUETOOTH:CONSTELLATION:TRACE1:FREEZE 1` freezes the updates of Trace 1.

[SENSe]:BLUEtooth:CONSTellation:TRACE<x>:POINtsPerSymbol

Sets or returns the number of points per symbol (how many points to use between symbols) for the specified trace (Trace 1 or Trace 2) in the Bluetooth Constellation display.

Conditions

Measurement views: Bluetooth Constellation

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:CONSTellation:TRACE<x>:POINtsPerSymbol { ONE |TWO | FOUR | EIGHt | SIXTeen |
   THIRtytwo }
[SENSe]:BLUEtooth:CONSTellation:TRACE<x>:POINtsPerSymbol?
```

Arguments

The number of points per symbol (1,2,4,8,16,32).

Examples

`SENSe:BLUETOOTH:CONSTELLATION:TRACE2:POINTSPERSYMBOL FOUR` sets the number of points per symbol for trace 2 to four.

[SENSe]:BLUEtooth:CONStellation:TRACe<x>:SHOW

Shows or hides the specified trace (Trace 1 or Trace 2) in the Bluetooth Constellation display.

Conditions

Measurement views: Bluetooth Constellation

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:CONStellation:TRACe<x>:SHOW { OFF | ON | 0 | 1 }
[SENSe]:BLUEtooth:CONStellation:TRACe<x>:SHOW?
```

Arguments

`OFF` or `0` hides the specified trace in the display.

`ON` or `1` shows the specified trace in the display.

Examples

`SENSe:BLUETOOTH:CONSTELLATION:TRACE1:SHOW OFF` hides trace 1 in the display.

[SENSe]:BLUEtooth:DELTa:AVERage:FONE

Sets or returns the average frequency for the low deviation pattern in the Bluetooth measurement.

Conditions

Measurement views: Any Bluetooth display

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:DELTa:AVERage:FONE <value>
[SENSe]:BLUEtooth:DELTa:AVERage:FONE?
```

Arguments

`<value> ::= <NR3>`, the average frequency of the low deviation pattern.

Examples

`SENSe:BLUETOOTH:DELTA:AVERAGE:FONE?` might return `139.8309531250E+3` indicating the average frequency of the low deviation pattern is 139.83 kHz.

[SENSe]:BLUEtooth:DELTa:AVERage:FTWO

Sets or returns the average frequency for the high deviation pattern in the Bluetooth measurement.

Conditions

Measurement views: Any Bluetooth display

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:DELTa:AVERage:FTWO <value>
[SENSe]:BLUEtooth:DELTa:AVERage:FTWO?
```

Arguments

`<value> ::= <NR3>`, the average frequency of the high deviation pattern.

Examples

`SENSe:BLUETOOTH:DELTA:AVERAGE:FTWO?` might return `139.8309531250E+3` indicating the average frequency of the low deviation pattern is 139.83 kHz.

[SENSe]:BLUEtooth:FDVTime:TRACE:OCTEt? (Query Only)

Returns the selected octet number on the Frequency Deviation vs. Time display.

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:FDVTime:TRACE:OCTEt?
```

Arguments

None

Returns

`<NR3>` the octet number.

Examples

`SENSe:BLUEtooth:FDVTime:TRACE:OCTEt?` might return `5` indicating that 5 is the selected octet number.

[SENSe]:BLUEtooth:FDVTime:TRACE:OCTEt:TOTAL:COUNt? (Query Only)

Returns the total octet value to be selected on the Frequency Deviation vs. Time display.

For the standard low-energy range the maximum number of octets available is 36. For standard basic rate range, the maximum number of octets depends on the packet type.

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:FDVTime:TRACE:OCTEt:TOTAL:COUNt?
```

Arguments

None

Returns

An <NR3> number indicating the maximum numbers of octets available.

Examples

`SENSe:BLUEtooth:FDVTime:TRACE:OCTEt:TOTAL:COUNt?` might return `36` indicating the maximum number of octets selected is 36.

[SENSe]:BLUEtooth:FDVTime:TRACE:VIEW:MODE

Sets or queries view mode in the Frequency Deviation vs. Time display.

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:FDVTime:TRACE:VIEW:MODE {PACK | OCTET}
[SENSe]:BLUEtooth:FDVTime:TRACE:VIEW:MODE?
```

Arguments

`PACK` indicates the display is in Packet mode.

`OCTET` indicates the display is in Octet mode.

Examples

`SENSe:BLUEtooth:FDVTime:TRACE:VIEW:MODE PACK` sets the view to packet mode.

[SENSe]:BLUEtooth:FILTer:ALPHa

Sets or returns the alpha filter factor (α/BT) in the Bluetooth analysis.

Conditions

Measurement views: Any Bluetooth display

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:FILTer:ALPHa <value>
[SENSe]:BLUEtooth:FILTer:ALPHa?
```

Arguments

`<value> ::= <NRf>` the filter factor; range: 0.001 to 1.

Examples

`SENSe:BLUETOOTH:FILTER:ALPHA 0.5` sets the filter factor to 0.5.

[SENSe]:BLUEtooth:FILTer:MEASurement

Sets or returns the measurement filter in the Bluetooth measurements.

Conditions

Measurement views: Any Bluetooth display

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:FILTer:MEASurement {OFF | BR | LE | USER1 | USER2| UOTH }
[SENSe]:BLUEtooth:FILTer:MEASurement?
```

Arguments

`OFF` no measurement filter is used.

`BR`, the Basic Rate filter is used.

`LE`, the Low Energy filter is used.

`USER1`, the User Meas Filter 1 is used.

`USER2`, the User Meas Filter 2 is used.

`UOTH` a user-defined filter is used.

Examples

`SENSe:BLUETOOTH:FILTER:MEASUREMENT USER1` uses the User Measurement Filter 1 as the measurement filter.

[SENSe]:BLUEtooth:FILTer:REFerence

Sets or returns the reference filter used in the Bluetooth measurements.

Conditions

Measurement views: Any Bluetooth display

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:FILTer:REFerence {OFF | BR | USER1 | USER2| UOTH }
[SENSe]:BLUEtooth:FILTer:REFerence?
```

Related Commands

AADD

Arguments

`OFF` no reference measurement filter is used.

`BR`, the Basic Rate filter is used.

`USER1`, the User Ref Filter 1 is used.

`USER2`, the User Ref Filter 2 is used.

`UOTH`, a user-defined filter is used.

Examples

`SENSe:BLUETOOTH:FILTER:REFERENCE OFF` selects None or no reference filter.

[SENSe]:BLUEtooth:FREQuency:DEViation

Sets or returns the frequency deviation in the Bluetooth analysis.

Conditions

Measurement views: Any Bluetooth display

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:FREQuency:DEViation <value>
[SENSe]:BLUEtooth:FREQuency:DEViation?
```

Related Commands

[SENSe]:BLUEtooth:FREQuency:DEViation:AUTO (on page 1999)

Arguments

`<value> ::= <NRf>`, the frequency deviation.

The Standard product range is 1 Hz to 40 MHz; the Option 110 product range is 1 Hz to 110 MHz.

Examples

`SENSe:BLUETOOTH:FREQUENCY:DEVIATION 1MHz` sets the frequency deviation to 1 MHz.

[SENSe]:BLUEtooth:FREQuency:DEViation:AUTO

Determines whether to detect the frequency deviation automatically or manually for the Bluetooth measurement.

Conditions

Measurement views: Any Bluetooth display

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:FREQuency:DEViation:AUTO { 0 | 1 | OFF | ON }
[SENSe]:BLUEtooth:FREQuency:DEViation:AUTO?
```

Related Commands

[SENSe]:BLUEtooth:FREQuency:DEViation (on page 1999)

Arguments

`OFF` or `0` sets the frequency deviation manually.

`ON` or `1` automatically calculates the frequency deviation; (default setting).

Examples

SENSe:BLUETOOTH:FREQUENCY:DEVIATION:AUTO ON automatically calculates the frequency deviation.

[SENSe]:BLUEtooth:FREQuency:ERRor

Sets or returns the frequency error in the Bluetooth analysis parameters (when Auto is off).

Conditions

Measurement views: Any Bluetooth display

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:FREQuency:ERRor <value>
[SENSe]:BLUEtooth:FREQuency:ERRor?
```

Related Commands

[SENSe]:BLUEtooth:FREQuency:ERRor:AUTO (on page 2000)

Arguments

<value> ::= <NRf>, the frequency error in Hz.

Examples

SENSe:BLUETOOTH:FREQUENCY:ERROR 0.02 sets the frequency error to 0.02.

[SENSe]:BLUEtooth:FREQuency:ERRor:AUTO

Determines whether the frequency error is set to Auto or Manual.

To enter a manual value, when Auto is Off, use the [SENSe]:BLUEtooth:FREQuency:ERRor command.

Conditions

Measurement views: Any Bluetooth display

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:FREQuency:ERRor:AUTO { 0 | 1 | OFF | ON }
[SENSe]:BLUEtooth:FREQuency:ERRor:AUTO?
```

Arguments

OFF or 0 sets the frequency error manually using the [SENSe]:BLUEtooth:FREQuency:ERRor command.

ON or 1 sets the frequency error automatically.

Examples

SENSe:BLUETOOTH:FREQUENCY:ERROR:AUTO ON determines the frequency error automatically.

[SENSe]:BLUEtooth:INBEmissions:POWEr:LIMIt<x>

Sets or queries the limits of the Bluetooth InBand Emission settings.

When <x> = 1, then commands set or query the fTX ± 2 MHz limit.

When <x> = 2, then commands set or query the fTX ± (3 + n) MHz limit.

Conditions

Measurement views: Bluetooth InBand Emissions display

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:INBEmissions:POWEr:LIMIt<x> <value>
[SENSe]:BLUEtooth:INBEmissions:POWEr:LIMIt<x>
```

Arguments

<value> ::= <NR3> the limit of the specified inband emissions setting in dBm.

Examples

SENSe:BLUETOOTH:INBEMISSIONS:POWER:LIMIT2? might return -20 indicating the fTX ± 2 limit is -20 dBm.

[SENSe]:BLUEtooth:POWer:CLASs

Sets or queries the power class of the Bluetooth standard.

Conditions

Measurement views: Bluetooth displays

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:POWer:CLASs {THREE|TWO|ONE}
[SENSe]:BLUEtooth:POWer:CLASs?
```

Arguments

THREE is power class 3.

TWO is power class 2.

ONE is power class 1.

Examples

SENSe:BLUETOOTH:POWER:CLASS THREE sets the power class to 3.

[SENSe]:BLUEtooth:STANdard

Sets or queries the Bluetooth standard.

Conditions

Measurement views: Bluetooth Constellation

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:STANdard {BR|LE|EDR}
[SENSe]:BLUEtooth:STANdard?
```

Arguments

`BR` is the Basic Rate standard.

`LE` is the Low Energy standard.

`EDR` is the Eye Diagram standard.

Examples

`SENSe:BLUETOOTH:STANDARD BR` sets the standard type to Basic Rate.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:DELTa:FONE:AVERage:MAX

Sets or returns the maximum limit of the Basic Rate ΔF1 average in the Bluetooth Constellation Summary display.

Conditions

Measurement views: Bluetooth summary

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:DELTa:FONE:AVERage:MAX<value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:DELTa:FONE:AVERage:MAX?
```

Arguments

`<value> ::= <NRf>` the maximum average limit in Hz.

Examples

`SENSe:BLUETOOTH:SUMMARY:LIMIT:BDR:DELTA:FONE:AVERAGE:MAX 175.0E+3` sets the maximum Basic Rate limit of the ΔF1 average to 175.0 kHz.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:DELTa:FONE:AVERage:MIN

Sets or returns the minimum limit of the Basic Rate ΔF1 average in the Bluetooth Constellation Summary display.

Conditions

Measurement views: Bluetooth summary

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:DELTa:FONE:AVERage:MIN<value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:DELTa:FONE:AVERage:MIN?
```

Arguments

<value> ::= <NRf> the minimum average limit in Hz.

Examples

SENSe:BLUETOOTH:SUMMARY:LIMIT:BDR:DELTA:FONE:AVERAGE:MIN 140.0E+3 sets the minimum limit of the Basic Rate ΔF1 average to 140.0 kHz.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:DELTa:FTWO:PERcent

Sets or returns the limit of the Basic Rate ΔF2 percentage in the Bluetooth Summary display.

Conditions

Measurement views: Bluetooth summary

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:DELTa:FTWO:PERcent<value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:DELTa:FTWO:PERcent?
```

Arguments

<value> ::= <NRf> the limit in percent (%).

Examples

SENSe:BLUETOOTH:SUMMARY:LIMIT:BDR:DELTA:FTWO:PERCENT? might return 99.00 indicating the limit of the Basic Rate ΔF2 percentage is 99%.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:DELTa:FTWO:RESult

Sets or returns the limit of the Basic Rate ΔF2 result in the Bluetooth Constellation Summary display.

Conditions

Measurement views: Bluetooth summary

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:DELTa:FTWO:RESult<value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:DELTa:FTWO:RESult?
```

Arguments

<value> ::= <NRf> the limit in Hz.

Examples

SENSe:BLUETOOTH:SUMMARY:LIMIT:BDR:DELTA:FTWO:RESULT 115.0E+3 sets the limit of the Basic Rate ΔF2 result to 115.0 kHz.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:MAX:DRIFt:FNFZero

Sets or returns the limit of the Basic Rate maximum drift fn–f0 in the Bluetooth Summary display.

Conditions

Measurement views: Bluetooth summary

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:MAX:DRIFt:FNFZero<value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:MAX:DRIFt:FNFZero?
```

Arguments

`<value> ::= <NRf>` the limit in Hz.

Examples

`SENSe:BLUETOOTH:SUMMARY:LIMIT:BDR:MAX:DRIFT:FNFZERO?` might return 20.00E+3 indicating the limit of the Basic Rate maximum drift fn–f0 is 20 kHz.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:MAX:FREQuency:OFFSet

Sets or returns the limit of the Basic Rate maximum frequency offset in the Bluetooth Summary display.

Conditions

Measurement views: Bluetooth summary

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:MAX:FREQuency:OFFSet <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:MAX:FREQuency:OFFSet?
```

Arguments

`<value> ::= <NRf>` the limit in Hz.

Examples

`SENSe:BLUETOOTH:SUMMARY:LIMIT:BDR:MAX:FREQUENCY:OFFSET 150.0E+3` sets the Basic Rate maximum frequency offset limit to 150 kHz.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:MAX:FREQuency:OFFSet:PREAmBDR

Sets or returns the limit of the Basic Rate maximum frequency offset value from the preamble region in the Bluetooth Summary display.

Conditions

Measurement views: Bluetooth summary

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:MAX:FREQuency:OFFSet:PREAmBDR <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:MAX:FREQuency:OFFSet:PREAmBDR?
```

Arguments

`<value> ::= <NRf>` the limit in Hz.

Examples

`SENSe:BLUETOOTH:SUMMARY:LIMIT:BDR:MAX:FREQUENCY:OFFSET: PREAMBDR 150.0E+3` sets the limit of the Basic Rate maximum frequency offset limit to 150 kHz.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:ONE:AVERage:MAX

Sets or returns the limit of the Basic Rate Power Class 1 maximum average output power value in the Bluetooth Summary display.

Conditions

Measurement views: Bluetooth summary

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:ONE:AVERage:MAX <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:ONE:AVERage:MAX?
```

Arguments

`<value> ::= <NRf>` the limit in dBm.

Examples

`SENSe:BLUETOOTH:SUMMARY:LIMIT:BDR:OPOWER:CLASS:ONE:AVERAGE:MAX?` might return `20.00` indicating the maximum output power average is 20.00 dBm.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:ONE:AVERage:MIN

Sets or returns the limit of the Basic Rate Power Class 1 minimum average output power value in the Bluetooth Summary display.

Conditions

Measurement views: Bluetooth summary

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:ONE:AVERage:MIN <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:ONE:AVERage:MIN?
```

Arguments

<value> ::= <NRf> the limit in dBm.

Examples

SENSe:BLUETOOTH:SUMMARY:LIMIT:BDR:OPOWER:CLASS:ONE:AVERAGE:MIN? might return 0.00 indicating the minimum output power average is 0 dBm.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:ONE:PEAK:MAX

Sets or returns the limit of the Basic Rate Power Class 1 maximum peak output power value in the Bluetooth Constellation Summary display.

Conditions

Measurement views: Bluetooth summary

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:ONE:PEAK:MAX <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:ONE:PEAK:MAX?
```

Arguments

<value> ::= <NRf> the limit in dBm.

Examples

SENSe:BLUETOOTH:SUMMARY:LIMIT:BDR:OPOWER:CLASS:ONE:PEAK:MAX? might return 23.00 indicating the maximum peak output power is 23.00 dBm.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:ONE:PEAK:MIN

Sets or returns the limit of the Basic Rate Power Class 1 minimum peak output power value in the Bluetooth Summary display.

Conditions

Measurement views: Bluetooth summary

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:ONE:PEAK:MIN <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:ONE:PEAK: MIN?
```

Arguments

<value> ::= <NRf> the limit in dBm.

Examples

SENSe:BLUETOOTH:SUMMARY:LIMIT:BDR:OPOWER:CLASS:ONE:PEAK:MIN? might return 3.00 indicating the maximum peak output power is 3.00 dBm.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:THREe:AVERage:MAX

Sets or returns the limit of the Basic Rate Power Class 3 maximum average output power value in the Bluetooth Summary display.

Conditions

Measurement views: Bluetooth summary

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:THREe:AVERage:MAX <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:THREe:AVERage:MAX?
```

Arguments

<value> ::= <NRf> the limit in dBm.

Examples

SENSe:BLUETOOTH:SUMMARY:LIMIT:BDR:OPOWER:CLASS:THREE:AVERAGE:MAX? might return -40.00 indicating the maximum output power average is -40.00 dBm.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:THREe:AVERage:MIN

Sets or returns the limit of the Basic Rate Power Class 3 minimum average output power value in the Bluetooth Constellation Summary display.

Conditions

Measurement views: Bluetooth summary

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:THREe:AVERage:MIN <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:THREe:AVERage:MIN?
```

Arguments

<value> ::= <NRf> the limit in dBm.

Examples

SENSe:BLUETOOTH:SUMMARY:LIMIT:BDR:OPOWER:CLASS:THREE:AVERAGE:MIN? might return 0.00 indicating the minimum output power average is 0.00 dBm.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:THREe:PEAK:MAX

Sets or returns the limit of the Basic Rate Power Class 3 maximum peak output power value in the Bluetooth Constellation Summary display.

Conditions

Measurement views: Bluetooth summary

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:THREe:PEAK:MAX <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:THREe:PEAK:MAX?
```

Arguments

`<value> ::= <NRf>` the limit in dBm.

Examples

`SENSe:BLUETOOTH:SUMMARY:LIMIT:BDR:OPOWER:CLASS:THREE:PEAK:MAX?` might return 37.00 indicating the maximum peak output power is 37.00 dBm.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:THREe:PEAK:MIN

Sets or returns the limit of the Basic Rate Power Class 3 minimum peak output power value in the Bluetooth Constellation Summary display.

Conditions

Measurement views: Bluetooth summary

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:THREe:PEAK:MIN <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:THREe:PEAK:MIN?
```

Arguments

`<value> ::= <NRf>` the limit in dBm.

Examples

`SENSe:BLUETOOTH:SUMMARY:LIMIT:BDR:OPOWER:CLASS:THREE:PEAK:MIN?` might return 3.00 indicating the maximum peak output power is 3.00 dBm.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:TWO:AVERage:MAX

Sets or returns the limit of the Basic Rate Power Class 2 maximum average output power value in the Bluetooth Constellation Summary display.

Conditions

Measurement views: Bluetooth summary

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:TWO:AVERage:MAX <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:TWO:AVERage:MAX?
```

Arguments

`<value> ::= <NRf>` the limit in dBm.

Examples

`SENSe:BLUETOOTH:SUMMARY:LIMIT:BDR:OPOWER:CLASS:TWO:AVERAGE:MAX?` might return 4.00 indicating the maximum output power average is 4.00 dBm.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:TWO:AVERage:MIN

Sets or returns the limit of the Basic Rate Power Class 2 minimum average output power value in the Bluetooth Constellation Summary display.

Conditions

Measurement views: Bluetooth summary

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:TWO:AVERage:MIN <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:TWO:AVERage:MIN?
```

Arguments

`<value> ::= <NRf>` the limit in dBm.

Examples

`SENSe:BLUETOOTH:SUMMARY:LIMIT:BDR:OPOWER:CLASS:TWO:AVERAGE:MIN?` might return -6.00 indicating the minimum output power average is -6.00 dBm.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:TWO:PEAK:MAX

Sets or returns the limit of the Basic Rate Power Class 2 maximum peak output power value in the Bluetooth Constellation Summary display.

Conditions

Measurement views: Bluetooth summary

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:TWO:PEAK:MAX <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:TWO:PEAK:MAX?
```

Arguments

`<value> ::= <NRf>` the limit in dBm.

Examples

`SENSe:BLUETOOTH:SUMMARY:LIMIT:BDR:OPOWER:CLASS:TWO:PEAK:MAX?` might return 7.00 indicating the maximum peak output power is 7.00 dBm.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:TWO:PEAK:MIN

Sets or returns the limit of the Basic Rate Power Class 2 minimum peak output power value in the Bluetooth Constellation Summary display.

Conditions

Measurement views: Bluetooth summary

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:TWO:PEAK:MIN <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:TWO:PEAK:MIN?
```

Arguments

`<value> ::= <NRf>` the limit in dBm.

Examples

`SENSe:BLUETOOTH:SUMMARY:LIMIT:BDR:OPOWER:CLASS:TWO:PEAK:MIN?` might return -3.00 indicating the maximum peak output power is -3.00 dBm.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:DELTa:FONE:AVERage:MIN

Sets or returns the minimum limit of the Low Energy ΔF1 average in the Bluetooth Constellation Summary display.

Conditions

Measurement views: Bluetooth summary

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:DELTa:FONE:AVERage:MIN <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:DELTa:FONE:AVERage:MIN?
```

Arguments

<value> ::= <NRf> the minimum average limit in Hz.

Examples

SENSe:BLUETOOTH:SUMMARY:LIMIT:BLE:DELTA:FONE:AVERAGE:MIN 225.0E+3 sets the minimum limit of the Low Energy ΔF1 average to 225.0 kHz.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:DELTa:FTWO:PERcent

Sets or returns the limit of the Low Energy ΔF2 percentage in the Bluetooth Constellation Summary display.

Conditions

Measurement views: Bluetooth summary

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:DELTa:FTWO:PERcent <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:DELTa:FTWO:PERcent?
```

Arguments

<value> ::= <NRf> the limit in percent (%).

Examples

SENSe:BLUETOOTH:SUMMARY:LIMIT:BLE:DELTA:FTWO:PERCENT? might return 99.00 indicating the limit of the Low Energy ΔF2 percentage is 99%.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:DELTa:FTWO:RESult

Sets or returns the limit of the Low Energy ΔF2 result in the Bluetooth Constellation Summary display.

Conditions

Measurement views: Bluetooth summary

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:DELTa:FTWO:RESult <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:DELTa:FTWO:RESult?
```

Arguments

<value> ::= <NRf> the limit in Hz.

Examples

SENSe:BLUETOOTH:SUMMARY:LIMIT:BLE:DELTA:FTWO:RESULT 185.0E+3 sets the limit of the Low Energy ΔF2 result to 185.0 kHz.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:MAX:DRIFt:FNFZero

Sets or returns the limit of the Low Energy maximum drift fn–f0 in the Bluetooth Constellation Summary display.

Conditions

Measurement views: Bluetooth summary

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:MAX:DRIFt:FNFZero <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:MAX:DRIFt:FNFZero?
```

Arguments

`<value> ::= <NRf>` the limit in Hz.

Examples

`SENSe:BLUETOOTH:SUMMARY:LIMIT:BLE:MAX:DRIFT:FNFZERO?` might return 50.0E+3 indicating the limit of the Low Energy maximum drift fn–f0 is 50 kHz.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:MAX:FREQuency:OFFSet

Sets or returns the limit of the Low Energy maximum frequency offset in the Bluetooth Constellation Summary display.

Conditions

Measurement views: Bluetooth summary

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:MAX:FREQuency:OFFSet <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:MAX:FREQuency:OFFSet?
```

Arguments

`<value> ::= <NRf>` the limit in Hz.

Examples

`SENSe:BLUETOOTH:SUMMARY:LIMIT:BLE:MAX:FREQUENCY:OFFSET 75.0E+3` sets the Basic Rate maximum frequency offset limit to 75 kHz.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:MAX:FREQuency:OFFSet:PREAmble

Sets or returns the limit of the Low Energy maximum frequency offset value from the preamble region in the Bluetooth Constellation Summary display.

Conditions

Measurement views: Bluetooth summary

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:MAX:FREQuency:OFFSet:PREAmble <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:MAX:FREQuency:OFFSet:PREAmble?
```

Arguments

`<value> ::= <NRf>` the limit in Hz.

Examples

`SENSe:BLUETOOTH:SUMMARY:LIMIT:BLE:MAX:FREQUENCY:OFFSET:PREAMBLE75.0E+3` sets the limit of the Low Enegy maximum frequency offset limit to 75 kHz.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:OPOWer:AVERage:MAX

Sets or returns the limit of the Low Energy maximum average output power value in the Bluetooth Constellation Summary display.

Conditions

Measurement views: Bluetooth summary

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:OPOWer:AVERage:MAX <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:OPOWer:AVERage:MAX?
```

Arguments

`<value> ::= <NRf>` the limit in dBm.

Examples

`SENSe:BLUETOOTH:SUMMARY:LIMIT:BLE:OPOWER:AVERAGE:MAX?` might return 10.00 indicating the Low Energy maximum output power average is 10.00 dBm.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:OPOWer:AVERage:MIN

Sets or returns the limit of the Low Energy minimum average output power value in the Bluetooth Constellation Summary display.

Conditions

Measurement views: Bluetooth summary

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:OPOWer:AVERage:MIN <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:OPOWer:AVERage:MIN?
```

Arguments

`<value> ::= <NRf>` the limit in dBm.

Examples

SENSe:BLUETOOTH:SUMMARY:LIMIT:BLE:OPOWER:AVERAGE:MIN? might return -20.00 indicating the Low Energy minimum output power average is -20 dBm.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:OPOWer:PEAK:MAX

Sets or returns the limit of the Low Energy maximum peak output power value in the Bluetooth Constellation Summary display.

Conditions

Measurement views: Bluetooth summary

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:OPOWer:PEAK:MAX <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:OPOWer:PEAK:MAX?
```

Arguments

<value> ::= <NRf> the limit in dBm.

Examples

SENSe:BLUETOOTH:SUMMARY:LIMIT:BLE:OPOWER:PEAK:MAX? might return 13.00 indicating the maximum peak output power is 13.00 dBm.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:DELTa:FONE:AVERage:MAX

Sets or returns the maximum limit of the Low Energy ΔF1 average in the Bluetooth Constellation Summary display.

Conditions

Measurement views: Bluetooth summary

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:DELTa:FONE:AVERage:MAX <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:DELTa:FONE:AVERage:MAX?
```

Arguments

<value> ::= <NRf> the maximum average limit in Hz.

Examples

SENSe:BLUETOOTH:SUMMARY:LIMIT:BLE:DELTA:FONE:AVERAGE:MAX 275.0E+3 sets the maximum Low Energy limit of the Low Energy ΔF1 average to 275.0 kHz.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:OPOWer:PEAK:MIN

Sets or returns the limit of the Low Energy minimum peak output power value in the Bluetooth Constellation Summary display.

Conditions

Measurement views: Bluetooth summary

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:OPOWer:PEAK:MIN <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:OPOWer:PEAK:MIN?
```

Arguments

`<value> ::= <NRf>` the limit in dBm.

Examples

`SENSe:BLUETOOTH:SUMMARY:LIMIT:BLE:OPOWER:PEAK:MIN?` might return -17.00 indicating the maximum peak output power is -17.00 dBm.

[SENSe]:BLUEtooth:SUMMary:LIMIt:MAX:DRIFt:F1FZero

Sets or returns the limit of the maximum fn–fn0 drift in the packet from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:MAX:DRIFt:F1FZero <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:MAX:DRIFt:F1FZero?
```

Arguments

`<value> ::= <NRf>` the limit in Hz.

Examples

`SENSe:BLUETOOTH:SUMMARY:LIMIT:MAX:DRIFT:F1FZERO 20.0E+3` sets the maximum fn–fn0 drift in the packet to 20.00 kHz.

[SENSe]:BLUEtooth:SUMMary:LIMIt:MAX:DRIFt:FNFN5

Sets or returns the limit of the maximum fn–fn5 drift in the packet from the Bluetooth summary.

Conditions

Measurement views: Bluetooth summary

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:MAX:DRIFt:FNFN5 <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:MAX:DRIFt:FNFN5?
```

Arguments

<value> ::= <NRf> the limit in Hz.

Examples

SENSe:BLUETOOTH:SUMMARY:LIMIT:MAX:DRIFT:FNFN5 20.0E+3 sets the maximum fn–fn5 drift in the packet to 20.00 kHz.

[SENSe]:BLUEtooth:SUMMary:LIMIt:MAX:DRIFt:LECoded:F0FThree

Sets and queries the limit of the LE Coded maximum drift frequency value of F0F3 in the Bluetooth Summary display.

Conditions

Measurement views: Bluetooth summary display

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:MAX:DRIFt:LECoded:F0FThree <0 | 1>,<value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:MAX:DRIFt:LECoded:F0FThree?
```

Arguments

<value> ::= <NRf>

1 indicates to enable tick mark in “On” column

0 indicates to disable tick mark in “On” column

Returns

<NRf>

Examples

SENSe:BLUETOOTH:SUMMARY:LIMIT:MAX:DRIFT:LECODED:F0FTHREE 1,85000 sets the limit value to 85 kHz and the tick mark to enabled in the “On” column.

[SENSe]:BLUEtooth:SUMMary:LIMIt:MAX:DRIFt:LECoded:F1FFour

Sets and queries the limit of the LE Coded maximum drift frequency value of F1F4 in the Bluetooth Summary display.

Conditions

Measurement views: Bluetooth summary display

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:MAX:DRIFt:LECoded:F1FFour <0 | 1>,<value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:MAX:DRIFt:LECoded:F1FFour?
```

Arguments

<value> ::= <NRf>

1 indicates to enable tick mark in “On” column

0 indicates to disable tick mark in “On” column

Returns

<NRf>

Examples

SENSe:BLUETOOTH:SUMMARY:LIMIT:MAX:DRIFT:LECODED:F1FFOUR 1,35000 sets the limit value to 35 kHz and the tick mark to enabled in the “On” column.

[SENSe]:BLUEtooth:SUMMary:LIMIt:MAX:DRIFt:LECoded:FNFN3

Sets and queries the limit of the LE Coded maximum drift frequency value of FNFN3 in the Bluetooth Summary display.

Conditions

Measurement views: Bluetooth summary display

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:MAX:DRIFt:LECoded:FNFN3 <0 | 1>,<value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:MAX:DRIFt:LECoded:FNFN3?
```

Arguments

<value> ::= <NRf>

1 indicates to enable tick mark in “On” column

0 indicates to disable tick mark in “On” column

Returns

<NRf>

Examples

SENSe:BLUETOOTH:SUMMARY:LIMIT:MAX:DRIFT:LECODED:FNFN3 1,15000 sets the limit value to 15 kHz and the tick mark to enabled in the “On” column.

[SENSe]:BLUEtooth:SYMBol:POINts

Sets or returns the number of points per symbol (how many points to use between symbols) in the Bluetooth measurement.

Conditions

Measurement views: Any Bluetooth display.

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:SYMBol:POINts { ONE |TWO | FOUR | EIGHt | SIXTeen | THIRtytwo }
[SENSe]:BLUEtooth:SYMBol:POINts?
```

Arguments

The number of points per symbol.

Examples

`SENSe:BLUETOOTH:SYMBOL:POINTS FOUR` sets the number of points per symbol to four.

[SENSe]:BLUEtooth:TEST:PATTern

Sets or returns the test pattern type in the Bluetooth displays.

Conditions

Measurement views: Any Bluetooth display.

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:TEST:PATTern {LDEViation|HDEViation|TPOTHER}
[SENSe]:BLUEtooth:TEST:PATTern?
```

Related Commands

[SENSe]:BLUEtooth:TEST:PATTern:AUTO:DETEct (on page 2018)

Arguments

`LDEViation` indicates the low deviation test pattern.

`HDEViation` indicates the high deviation test pattern.

`TPOTHER` indicates a user defined test pattern.

Examples

`SENSe:BLUETOOTH:TEST:PATTERN?` might return LDEV indicating the selected test pattern type is the low deviation type.

[SENSe]:BLUEtooth:TEST:PATTern:AUTO:DETEct

Sets or returns the auto detect setting for the Bluetooth displays.

Conditions

Measurement views: Any Bluetooth display.

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:TEST:PATTern:AUTO:DETEct {0|1}
[SENSe]:BLUEtooth:TEST:PATTern:AUTO:DETEct?
```

Related Commands

[SENSe]:BLUEtooth:TEST:PATTern (on page 2018)

Arguments

1 indicates that the auto detect test pattern is enabled (checked in the view).

0 indicates that the auto detect test pattern is not enabled (not checked).

Examples

SENSe:BLUETOOTH:TEST:PATTERN:AUTO:DETECT? might return 0 indicating that the auto detect test pattern is not enabled.

[SENSe]:BLUEtooth:TIME:ZERO:REFERENCE

Sets or returns the Time Zero Reference setting in the Bluetooth display.

Conditions

Measurement views: Any Bluetooth display.

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:TIME:ZERO:REFERENCE {Acquisistion Start|Trigger}
[SENSe]:BLUEtooth:TIME:ZERO:REFERENCE?
```

Arguments

Acquisition Start sets the Time Zero Reference to the start of the acquisition.

Trigger sets the Time Zero Reference to the trigger position.

Examples

SENSe:BLUETOOTH:TIME:ZERO:REFERENCE? might return TRIGGER indicating the Time Zero Reference is the trigger position.

[SENSe]:BLUEtooth:TRACe:TYPE

Sets or queries the type of trace on the Trace tab of the control panel.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:TRACe:TYPE {IQ|FREQDEV}
[SENSe]:BLUEtooth:TRACe:TYPE?
```

Arguments

IQ specifies the IQ trace.

FREQDEV specifies the Frequency Deviation trace.

Examples

SENSe:BLUETOOTH:TRACE:TYPE? might return IQ indicating the Trace type is set to IQ.

[SENSe]:CONSte[:MEASview<y>]:SELEcted:SOURce

Sets or queries the Source of Constellation for a selected Measurement view number.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Constellation

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (ex: GPDemod Constellation).

Source cannot be changed when acquisition or analysis is in progress. Stop the acquisition and change the Source.

Group

Sense commands

Syntax

```
[SENSe]:CONSte[:MEASview<y>]:SELEcted:SOURce {"Source 1" | "Source 2" | "Source 3" | "Source 4"}
[SENSe]:CONSte[:MEASview<y>]:SELEcted:SOURce?
```

Arguments

The parameter `<y>`=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1.

Examples

`SENSe:CONSte:MEASview1:SELEcted:SOURce?` queries the Source for Constellation Measurement View 1.

[SENSe]:CCDF:{BANDwidth|BWIDth}

Sets or queries the CCDF measurement bandwidth (frequency span).

Conditions

Measurement views: CCDF

Group

Sense commands

Syntax

```
[SENSe]:CCDF:{BANDwidth|BWIDth} <value>
[SENSe]:CCDF:{BANDwidth|BWIDth}?
```

Arguments

`<value> ::= <NRf>` is the CCDF measurement bandwidth.

Range: 10 Hz to 40 MHz (Standard) / 60 MHz (Option 110).

Examples

`SENSe:CCDF:BANDWIDTH 1MHz` sets the CCDF measurement bandwidth to 1 MHz.

[SENSe]:CCDF:CLEar (No Query Form)

Clears the CCDF accumulator and restarts the measurement.

Conditions

Measurement views: CCDF

Group

Sense commands

Syntax

```
[SENSe]:CCDF:CLEar
```

Arguments

None

Examples

`SENSe:CCDF:CLEAR` clears the CCDF accumulator and restarts the measurement.

[SENSe]:CCDF:TIME:TOTal:LENGth

Sets or queries the CCDF measurement time when `[SENSe]:CCDF:TIME:TYPE` is set to TOTal.

Conditions

Measurement views: CCDF

Group

Sense commands

Syntax

```
[SENSe]:CCDF:TIME:TOTal:LENGth <value>
[SENSe]:CCDF:TIME:TOTal:LENGth?
```

Arguments

`<value>::<NRf>` specifies the CCDF measurement time.

Range: 20 ms to 100 s.

Examples

`SENSe:CCDF:TIME:TOTAL:LENGTH 10` sets the CCDF measurement time to 10 s.

[SENSe]:CCDF:TIME:TYPE

Determines how to repeat the CCDF measurement.

Conditions

Measurement views: CCDF

Group

Sense commands

Syntax

```
[SENSe]:CCDF:TIME:TYPE { SINGLe | TOTal | CONTinuous }
[SENSe]:CCDF:TIME:TYPE?
```

Related Commands

INITiate:CONTinuous (on page 1480)

INITiate[:IMMediate] (on page 1480)

INITiate:RESume (on page 1481)

Arguments

`SINGLe` specifies that the analyzer sets the analysis length to 1 ms and then acquire data once to calculate CCDF.

`TOTal` specifies that the analyzer sets the analysis length to 20 ms and then repeats data acquisition and CCDF calculation for the time specified by the [SENSe]:CCDF:TIME:TOTal:LENGth (on page 2021) command.

`CONTinuous` specifies that the analyzer sets the analysis length to 1 ms and then repeats data acquisition and CCDF calculation continuously. To reset the process, use the [SENSe]:CCDF:CLEar (on page 2020) command or the INITiate commands.

Examples

`SENSe:CCDF:TIME:TYPE SINGLe` specifies that the analyzer sets the analysis length to 1 ms and then acquire data once to calculate CCDF.

[SENSe]:DDEMod[:SOURce<x>]:ANALysis:LENGth

Sets or queries the analysis length in seconds or symbols. The command `[SENSe]:DDEMod[:SOURce<x>]:TIMe:UNITs` determines which is used. For example, if the symbol rate is 1 MHz and the acquisition length is 20 ms, the range may be 0 to 19999. Setting a specified length changes `[SENSe]:DDEMod[:SOURce<x>]:ANALysis:AUTO` to OFF.

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:ANALysis:LENGth<value>
[SENSe]:DDEMod[:SOURce<x>]:ANALysis:LENGth?
```

Related Commands

[SENSe]:DDEMod[:SOURce<x>]:ANALysis:LENGth:ACTual? (on page 2023)

[SENSe]:DDEMod[:SOURce<x>]:TIMe:UNITs (on page 2048)

[SENSe]:DDEMod[:SOURce<x>]:ANALysis:LENGth:AUTO (on page 2023)

Arguments

`<x>` = 1 to 8, representing Range of Source

`<value>::<NRf>` specifies the analysis length in seconds or symbols.

Range in seconds: 200 ns to `[(acquisition length) - 400 ns]`.

If `[(analysis start) + (analysis length)] > [(acquisition length) - 400 ns]`, the actual analysis length is reduced to `[(acquisition length) - 200 ns]`.

Range in symbols: `200 ns * (symbol rate) to [(acquisition length) - 400 ns] *(symbol rate)`.

If `[(analysis start) + (analysis length)] > [(acquisition length) - 400 ns]`, the actual analysis length is reduced to `[(acquisition length) - 200 ns] * (symbol rate)`.

Examples

`SENSe:DDEMOD:SOURce1:ANALYSIS:LENGTH 25.625µs` sets the analysis length of the measurement source1 to 25.625 µs.

`SENSe:DDEMOD:SOURce1:ANALYSIS:LENGTH 256` sets the analysis length of the measurement source1 to 256 symbols.

[SENSe]:DDEMod[:SOURce<x>]:ANALysis:LENGth:ACTual? (Query only)

Queries the actual analysis length which returns a value in either seconds or symbols. The command **[SENSe]:DDEMod[:SOURce<x>]:TIMe:UNITs** determines which is used.

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

`[SENSe]:DDEMod[:SOURce<x>]:ANALysis:LENGth:ACTual?`

Related Commands

[SENSe]:DDEMod[:SOURce<x>]:ANALysis:LENGth (on page 2022)

[SENSe]:DDEMod[:SOURce<x>]:TIMe:UNITs (on page 2048)

Arguments

`<x>` = 1 to 8, representing Range of Source.

Returns

`<NRf>` Actual analysis length in seconds or symbols.

Examples

`SENSe:DDEMOD:SOURCE1:ANALYSIS:LENGTH:ACTUAL?` might return `25.625E-6`, indicating that the actual analysis length of the measurement source1 is 25.625 µs.

[SENSe]:DDEMod[:SOURce<x>]:ANALysis:LENGth:AUTO

Determines whether to set the analysis length of the measurement source automatically or manually.

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:ANALysis:LENGth:AUTO { OFF | ON | 0 | 1 }
[SENSe]:DDEMod[:SOURce<x>]:ANALysis:LENGth:AUTO?
```

Arguments

<x> = 1 to 8, representing Range of Source.

OFF or 0 sets the analysis length manually, using the [SENSe]:DDEMod[:SOURce<x>]:ANALysis:LENGth (on page 2022) command.

ON or 1 sets the analysis length automatically.

Examples

SENSe:DDEMOD:SOURce1:ANALYSIS:LENGTH:AUTO ON sets the analysis length of the measurement source1 automatically.

[SENSe]:DDEMod[:SOURce<x>]:{BANDwidth|BWIDth}:MODE

Sets or queries the measurement bandwidth mode of the measurement source for Digital Demod measurements.

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:{BANDwidth|BWIDth}:MODE { MANual | AUTO | LSPan }
[SENSe]:DDEMod[:SOURce<x>]:{BANDwidth|BWIDth}:MODE?
```

Related Commands

[SENSe]:DDEMod[:SOURce<x>]:{BANDwidth|BWIDth}:TINTerval (on page 2024)

Arguments

<x> = 1 to 8, representing Range of Source.

MANual specifies manual bandwidth control mode. A typical use is setting a narrow measurement bandwidth on a small range of frequencies to examine one signal while rejecting the others that are present in a Spectrum graph.

AUTO specifies automatic bandwidth control mode where a measurement picks a bandwidth based on other parameter settings (such as symbol rate, modulation type, filter, and so on).

LSPan (LinkToSpan) specifies Lspan bandwidth control mode, where you can use the Spectrum display to tune the frequency, view the signal, and use the Span control to set the measurement bandwidth. This mode emulates legacy instruments.

Examples

SENSe:DDEMOD:SOURce1:BANDWIDTH:MODE AUTO sets the analyzer bandwidth mode of the measurement source1 to automatically select a measurement bandwidth.

[SENSe]:DDEMod[:SOURce<x>]:{BANDwidth|BWIDth}:TINTerval

Sets or queries the measurement bandwidth of the measurement source (frequency span).

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:{BANDwidth|BWIDth}:TINTerval<value>
[SENSe]:[:SOURce<x>]:{BANDwidth|BWIDth}:TINTerval?
```

Related Commands

[SENSe]:DDEMod[:SOURce<x>]:SRATe (on page 2043)

Arguments

`<x>` = 1 to 8, representing Range of Source.

`<value> ::= <NRf>` specifies the measurement bandwidth.

Range: Symbol rate to 40 MHz or with option 110, 110 MHz.

The minimum value depends on the setting of `[:SENSe]:DDEMod[:SOURce<x>]:SRATe`.

Returns

`<NRf>` Actual analysis length in seconds or symbols.

Examples

`SENSe:DDEMOD:SOURce1:BANDWIDTH:TINTERVAL 35.255MHz` sets the measurement bandwidth of the measurement source1 to 35.255 MHz.

[SENSe]:DDEMod[:SOURce<x>]:{BANDwidth|BWIDth}:TINTerval:AUTO

Sets or queries the measurement bandwidth of the measurement source (frequency span) automatically.

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:{BANDwidth|BWIDth}:TINTerval:AUTO { OFF | ON| 0 | 1 }
[SENSe]:DDEMod[:SOURce<x>]:{BANDwidth|BWIDth}:TINTerval:AUTO?
```

Related Commands

[SENSe]:DDEMod[:SOURce<x>]:{BANDwidth|BWIDth}:TINTerval (on page 2024)

Arguments

`<x>` = 1 to 8, representing Range of Source.

OFF or 0 sets the measurement bandwidth manually, using the [SENSe]:DDEMod[:SOURce<x>]:{BANDwidth|BWIDth}:TINTerval command.

ON or 1 sets the measurement bandwidth automatically.

Examples

SENSe:DDEMOD:SOURce1:BANDWIDTH:TINTERVAL:AUTO sets the measurement bandwidth of the measurement source1 automatically.

[SENSe]:DDEMod[:SOURce<x>]:BURSt:DETect

Sets or queries burst detection of the measurement source.

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:BURSt:DETect { ON | OFF }
[SENSe]:DDEMod[:SOURce<x>]:BURSt:DETect?
```

Related Commands

[SENSe]:DDEMod[:SOURce<x>]:BURSt:THReshold (on page 2026)

Arguments

<x> = 1 to 8, representing Range of Source.

ON analyzes just that burst period if a burst is found. If a burst is not found, does not analyze but displays an error message.

OFF analyzes the whole analysis length.

NOTE: When selecting On and if the signal is not adequate for the demodulation, the measurement will fail and show an error message.

Examples

SENSe:DDEMOD:SOURCE1:BURST:DETECT OFF disables burst detection of the measurement source and analyzes the whole analysis length.

[SENSe]:DDEMod[:SOURce<x>]:BURSt:THReshold

Sets or queries the threshold level of the measurement source above which the input signal is determined to be a burst.

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:BURSt:THReshold<value>
[SENSe]:DDEMod[:SOURce<x>]:BURSt:THReshold?
```

Related Commands

[SENSe]:DDEMod[:SOURce<x>]:BURSt:DETect (on page 2026)

Arguments

<x> = 1 to 8, representing Range of Source

<value>::<NRf> specifies the threshold level for detecting bursts.

Range: -100 to -10 dBc.

Examples

SENSe:DDEMOD:SOURce1:BURST:THRESHOLD -25 sets the threshold level of the measurement source1 to -25 dBc.

[SENSe]:DDEMod[:SOURce<x>]:CARRier:OFFSet

Sets or queries the carrier frequency offset for the measurement source in digital modulation analysis.

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:CARRier:OFFSet<value>
[SENSe]:DDEMod[:SOURce<x>]:CARRier:OFFSet?
```

Related Commands

The settings of the following commands may narrow the effective range.

[SENSe]:DDEMod[:SOURce<x>]:FILTer:ALPHa (on page 2032)

[SENSe]:DDEMod[:SOURce<x>]:FILTer:REFerence (on page 2033)

[SENSe]:DDEMod[:SOURce<x>]:MODulation:TYPE (on page 2037)

[SENSe]:DDEMod[:SOURce<x>]:SRATe (on page 2043)

Arguments

<x> = 1 to 8, representing Range of Source.

<value> ::= <NRf> specifies the carrier frequency offset.

Range: -20 MHz to +20 MHz or with Option 110, -55 MHz to +55 MHz.

Examples

SENSe:DDEMOD:SOURCE1:CARRIER:OFFSET 2kHz sets the carrier frequency offset to 2 kHz for the measurement source 1.

[SENSe]:DDEMod[:SOURce<x>]:CARRier:OFFSet:AUTO

Sets or queries whether to detect the carrier frequency automatically for the measurement source in digital modulational analysis.

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:CARRier:OFFSet:AUTO { OFF | ON | 0 | 1 }
[SENSe]:DDEMod[:SOURce<x>]:CARRier:OFFSet:AUTO?
```

Related Commands

[SENSe]:DDEMod[:SOURce<x>]:CARRier:OFFSet (on page 2027)

Arguments

<x> = 1 to 8, representing Range of Source

OFF or 0 sets the carrier frequency manually, using the [SENSe]:DDEMod[:SOURce<x>]:CARRier:OFFSet command.

ON or 1 detects the carrier frequency automatically.

Examples

SENSe:DDEMOD:SOURce1:CARRIER:OFFSET:AUTO ON enables automatic detection of the carrier frequency for measurement source1.

[SENSe]:DDEMod[:SOURce<x>]:EQUalizer:AVAilable? (Query only)

Determines if the equalizer is available for the measurement source.

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:EQUalizer:AVAilable?
```

Arguments

<x> = 1 to 8, representing Range of Source.

Returns

0 indicates the equalizer is not available for the current modulation type and reference filter.

1 indicates the equalizer is available.

[SENSe]:DDEMod[:SOURce<x>]:EQUalizer:CONVergence

Sets or queries the Convergence value (also known as tap update rate) of the measurement source.

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:EQUalizer:CONVergence <value>
[SENSe]:DDEMod[:SOURce<x>]:EQUalizer:CONVergence?
```

Arguments

`<x>` = 1 to 8, representing Range of Source.

`<value>::<NR3>` specifies the convergence value.

Examples

`SENSe:DDEMOD:SOURce1:EQUALIZER:CONVERGENCE?` might return `1.0000000000E-6` for measurement source 1.

[SENSe]:DDEMod[:SOURce<x>]:EQUalizer:ENABle

Sets or queries the Equalizer state of the measurement source.

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:EQUalizer:ENABle {OFF | ON | 0 | 1 }
[SENSe]:DDEMod[:SOURce<x>]:EQUalizer:ENABle?
```

Related Commands

[SENSe]:DDEMod[:SOURce<x>]:EQUalizer:MODE (on page 2030)

Arguments

`<x>` = 1 to 8, representing Range of Source.

`ON` or `1` enables the Equalizer.

`OFF` or `0` disables the Equalizer.

Examples

`SENSe:DDEMod:SOURCE1:EQU:ENAB 1` might return `1`, indicating that the equalizer is enabled for the measurement source1.

[SENSe]:DDEMod[:SOURce<x>]:EQUalizer:LENGth

Sets or queries the equalizer filter length of measurement source.

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:EQUalizer:LENGth<value>
```

Arguments

<x> = 1 to 8, representing Range of Source.

<value>::<NR1> specifies the equalizer filter length.

Examples

SENSe:DDEMOD:SOURce1:EQUALIZER:LENGTH 39 sets the filter length of measurement source 1 to 39 symbols.

[SENSe]:DDEMod[:SOURce<x>]:EQUalizer:MODE

Sets or queries the equalizer mode of the measurement source. The equalizer can in learning mode (train) or enabled (hold).

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:EQUalizer:MODE {TRAIn | HOLD }
[SENSe]:DDEMod[:SOURce<x>]:EQUalizer:MODE?
```

Related Commands

[SENSe]:DDEMod[:SOURce<x>]:EQUalizer:ENABle (on page 2029)

Arguments

<x> = 1 to 8, representing Range of Source.

TRAIn places the equalizer in training mode.

HOLD places in the equalizer in enabled mode.

Examples

SENSe:DDEMOD:SOURce1:EQUALIZER:MODE HOLD places the equalizer in enabled mode for the measurement source1.

[SENSe]:DDEMod[:SOURce<x>]:EQUalizer:RESet (No query form)

Resets the equalizer settings of the measurement source to default values.

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:EQUalizer:RESet
```

Examples

`SENSe:DDEMOD:SOURce1:EQUALIZER:RESET` sets the equalizer parameters of the measurement source1 to their default values.

[SENSe]:DDEMod[:SOURce<x>]:EQUalizer:TAPS

Sets or queries the number of filter coefficients for the equalizer of the measurement source.

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:EQUalizer:TAPS <value>
[SENSe]:DDEMod[:SOURce<x>]:EQUalizer:TAPS?
```

Arguments

`<x>` = 1 to 8, representing Range of Source.

`<value>::=<NR1>` the filter coefficient for the equalizer.

Examples

`SENSe:DDEMOD:SOURce<x>:EQUALIZER:TAPS 65` sets the measurement source1 filter coefficient to 65.

[SENSe]:DDEMod[:SOURce<x>]:EQUalizer:TSRAtio

Sets or queries the Equalizer taps or Symbol parameter for the measurement source.

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:EQUalizer:TSRAtio { ONE | TWO | FOUR | EIGHt }
```

Arguments

Valid arguments are `ONE`, `TWO`, `FOUR`, or `EIGHt`. Numerals are not valid arguments.

`<x>` = 1 to 8, representing Range of Source.

Examples

`SENSe:DDEMOD:SOURCE1:EQUALIZER:TSRATIO FOUR` sets the Taps or Symbol value to four for the measurement source1.

[SENSe]:DDEMod[:SOURce<x>]:FILTer:ALPHa

Sets or queries the filter factor (α/BT) for the measurement source in the digital modulation analysis.

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:FILTer:ALPHa <value>
[SENSe]:DDEMod[:SOURce<x>]:FILTer:ALPHa?
```

Arguments

`<x>` = 1 to 8, representing Range of Source.

`<value>::<NRf>` specifies the filter factor. Range: 0.001 to 1.

Examples

`SENSe:DDEMOD:SOURCE1:FILTER:ALPHA 0.5` sets the filter factor to 0.5 for the measurement source1.

[SENSe]:DDEMod[:SOURce<x>]:FILTer:MEASurement

Sets or queries the measurement filter for the measurement source in the digital modulation analysis.

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:FILTer:MEASurement { OFF | RRCosine | RCOSine | GAUSsian | RECTangular |
    IS95TXEQ_MEA | IS95TX_MEA | C4FM_P25 | USERx | UOTHer }
[SENSe]:DDEMod[:SOURce<x>]:FILTer:MEASurement?
```

Arguments

The following table lists the arguments.

<x> = 1 to 8, representing Range of Source.

Digital modulation measurement filter

Argument	Measurement filter
OFF	No filter
RRCosine	Root Raised Cosine
RCOSine	Raised Cosine
GAUSsian	Gaussian
RECTangular	Rectangular
IS95TXEQ_MEA	IS95 receive filter for the transmitter configured with both the transmit filter and the phase equalizer.
IS95TX_MEA	IS95 receive filter for the transmitter configured with only the transmit filter.
C4FM_P25	C4FM-P25
USER1	User defined Measurement Filter 1
USER2	User defined Measurement Filter 2
USER3	User defined Measurement Filter 3
UOTHer	Other user defined Measurement Filter

Examples

`SENSe:DDEMOD:SOURCE1:FILTER:MEASUREMENT RRCosine` selects the Root Raised Cosine filter as the measurement filter for the measurement source1.

[SENSe]:DDEMod:SOURCE<x>:FILTer:REFerence

Sets or queries the reference filter for the measurement source in the digital modulation analysis..

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:FILTer:REFerence { OFF | RCOSine | GAUSsian | RECTangular | IS95REF |
   HSINe | SOQPSK_MIL | SOQPSK_ARTM | SBPSK_MIL | USERx | UOTHer }
[SENSe]:DDEMod[:SOURce<x>]:FILTer:REFerence?
```

Arguments

<x> = 1 to 8, representing Range of Source.

The following table lists the arguments.

Digital modulation reference filter

Argument	Measurement filter
OFF	No filter
RCOSine	Raised Cosine
GAUSsian	Gaussian
RECTangular	Rectangular
IS95REF	IS95 reference filter including the response of the transmit filter, the phase equalizer, and the receive (complementary) filter.
HSINe	Half Sine
SOQPSK_MIL	SOQPSK–MIL
SOQPSK_ARTM	SOQPSK–ARTM
SBPSK_MIL	SBPSK–MIL
USER1	User defined Measurement Filter 1
USER2	User defined Measurement Filter 2
USER3	User defined Measurement Filter 3
UOTHer	Other user defined Measurement Filter

Examples

`SENSe:DDEMOD:SOURCE1:FILTER:REFERENCE RCOSine` selects the Raised Cosine filter as the reference filter for the measurement source1.

[SENSe]:DDEMod[:SOURce<x>]:FREQuency:DEViation

Sets or queries the frequency deviation of the measurement source in the digital modulation analysis. Programming a specified frequency deviation sets `[SENSe]:DDEMod[:SOURce<x>]:FREQuency:DEViation:AUTO OFF`.

This command is valid when `[SENSe]:DDEMod[:SOURce<x>]:MODulation:TYPE` is set to C4FM, FSK2, FSK4, FSK8, or FSK16.

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:FREQuency:DEViation<value>
[SENSe]:DDEMod[:SOURce<x>]:FREQuency:DEViation?
```

Arguments

<x> = 1 to 8, representing Range of Source.

<value>::<NRf> sets the frequency deviation.

Standard product range: 1 Hz to 40 MHz.

Option 110 product range: 1 Hz to 110 MHz.

Examples

SENSe:DDEMOD:SOURce1:FREQUENCY:DEVIATION 1MHz sets the frequency deviation of the measurement source1 to 1 MHz.

[SENSe]:DDEMod[:SOURce<x>]:FREQuency:DEViation:AUTO

Determines whether to detect automatically or set manually the frequency deviation of the measurement source used to determine the symbol values of an FSK or C4FM signal.

This command is valid when [SENSe]:DDEMod[:SOURce<x>]:MODulation:TYPE is set to C4FM, FSK2, FSK4, FSK8, or FSK16.

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:FREQuency:DEViation:AUTO { OFF | ON | 0 | 1 }
[SENSe]:DDEMod[:SOURce<x>]:FREQuency:DEViation:AUTO?
```

Arguments

<x> = 1 to 8, representing Range of Source.

ON or 1 automatically calculates the frequency deviation for the analysis range (default).

OFF or 0 sets the frequency deviation using the [SENSe]:DDEMod[:SOURce<x>]:FREQuency:DEViation command.

Examples

SENSe:DDEMOD:SOURce1:FREQUENCY:DEVIATION:AUTO ON automatically calculates the frequency deviation of the measurement source1.

[SENSe]:DDEMod[:SOURce<x>]:MAGNitude:NORMalize

Sets or queries the method for the magnitude normalization of the measurement source.

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:MAGNitude:NORMalize { RSYMbol | MSYMbol }
[SENSe]:DDEMod[:SOURce<x>]:MAGNitude:NORMalize?
```

Arguments

<x> = 1 to 8, representing Range of Source

RSYMbol normalizes the magnitude with the RMS symbol magnitude.

MSYMbol normalizes the magnitude with the maximum symbol magnitude.

Examples

SENSe:DDEMOD:SOURCE1:MAGNITUDE:NORMALIZE RSYMbol normalizes the magnitude of measurement source1 with the RMS symbol magnitude.

[SENSe]:DDEMod[:SOURce<x>]:MINDex

Sets or queries the measurement source modulation index of a CPM signal.

This command is valid when [SENSe]:DDEMod[:SOURce<x>]:MODulation:TYPE is set to CPM and [SENSe]:DDEMod[:SOURce<x>]:MINDex:AUTO is set to OFF.

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:MINDex { 1 | 2 | 3 | 4 | 5 | 6 }
[SENSe]:DDEMod[:SOURce<x>]:MINDex?
```

Arguments

<x> = 1 to 8, representing Range of Source.

The following table shows the arguments and modulation index.

CPM modulation index

Argument	Modulation index
1	4/16, 5/16
2	5/16, 6/16
3	6/16, 7/16
4	7/16, 10/16

Argument	Modulation index
5	12/16, 13/16
6	8/16, 8/16

Examples

`SENSe:DDEMOD:SOURCE1:MINDEX 1` selects the measurement source1 modulation index to `"4/16, 5/16"`.

[SENSe]:DDEMod[:SOURce<x>]:MINDex:AUTO

Determines whether to detect automatically or set manually the measurement source modulation index of a CPM signal. This command is valid when `[SENSe]:DDEMod[:SOURce<x>]:MODulation:TYPE` is set to CPM.

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:MINDex:AUTO { OFF | ON | 0 | 1 }
[SENSe]:DDEMod[:SOURce<x>]:MINDex:AUTO
```

Arguments

`<x>` = 1 to 8, representing Range of Source.

`ON` or `1` automatically calculates the modulation index for the analysis range.

`OFF` or `0` sets the modulation index using the `[SENSe]:DDEMod[:SOURce<x>]:MINDex` command.

Examples

`SENSe:DDEMOD:SOURce1:MINDEX:AUTO ON` automatically calculates the measurement source1 modulation index for the analysis range.

[SENSe]:DDEMod[:SOURce<x>]:MODulation:TYPE

Sets or queries the modulation type for the measurement source in the digital modulation analysis.

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:MODulation:TYPE { QPSK | PSK8 | D8PSK | PIOVER2DBPSK | DQPSK |
   PIOVER4DQPSK | BPSK | OQPSK | QAM16 | QAM32 | QAM64 | QAM128 | QAM256 | QAM1024 | MSK | FSK2 |
   FSK4 | FSK8 | FSK16 | CPM | APSK16| APSK32 | SOQPSK | SBPSK | C4FM }
[SENSe]:DDEMod[:SOURce<x>]:MODulation:TYPE?
```

Arguments

<x> = 1 to 8, representing Range of Source.

The following table lists the arguments and corresponding modulation type.

Modulation type

Argument	Modulation type
QPSK	QPSK
PSK8	8PSK
D8PSK	D8PSK
PIOVER2DBPSK	π/2 DBPSK
DQPSK	DQPSK
PIOVER4DQPSK	π/4 QPSK
BPSK	BPSK
OQPSK	OQPSK
QAM16	16QAM
QAM32	32QAM
QAM64	64QAM
QAM128	128QAM
QAM256	256QAM
QAM1024	1024QAM
MSK	MSK
FSK2	FSK2
FSK4	FSK4
FSK8	FSK8
FSK16	FSK16
CPM	CPM
APSK16	APSK16
APSK32	APSK32

Argument	Modulation type
SOQPSK	SOQPSK
SBPSK	SBPSK
C4FM	C4FM

Examples

`SENSe:DDEMOD:SOURCE1:MODULATION:TYPE QPSK` selects QPSK modulation system for the measurement source1.

[SENSe]:DDEMod[:SOURce<x>]:PRESet (No query form)

Presets the modulation analysis analysis of the measurement source to a communication standard.

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

`[SENSe]:DDEMod[:SOURce<x>]:PRESet <standard_name>`

Related Commands

[SENSe]:DDEMod[:SOURce<x>]:MODulation:TYPE (on page 2037)

Arguments

<x> = 1 to 8, representing Range of Source.

The following table lists the preset standard names with their modulation types and settings.

Modulation Presets

Preset Name	Modulation type	Settings
“802.15.4”	OQPSK	Symbol rate: 1 MHz
		Meas Filter: None
		Ref Filter: Half sine
		Filter Param: None
		Symbol Pt Location: NA
		Remove Q offset: not checked

Preset Name	Modulation type	Settings
“SBPSK-MIL”	SBPSK	Symbol rate: 2.4 kHz
		Meas Filter: None
		Ref Filter: SBPSK-MIL
		Filter Param: 0.5
		Symbol Pt Location: NA
		Remove Q offset: NA
“SOQPSK-MIL”	SOQPSK	Symbol rate: 2.4 kHz
		Meas Filter: None
		Ref Filter: SOQPSK-MIL
		Filter Param: 0.5
		Symbol Pt Location: Center
		Remove Q offset: checked
“CPM-MIL”	CPM	Symbol rate: 19.2 KHz
		Meas Filter: None
		Ref Filter: None
		Filter Param: None
		Symbol Pt Location: NA
		Remove Q offset: NA
“SOQPSK-ARTM Tier 1”	SOQPSK	Symbol rate: 2.5 MHz
		Meas Filter: None
		Ref Filter: SOQPSK-ARTM
		Filter Param: None
		Symbol Pt Location: Center
		Remove Q offset: checked

Preset Name	Modulation type	Settings
“Project25 Phase I”	C4FM	Symbol rate: 4.8 kHz
		Meas Filter: C4FM-P25
		Ref Filter: Raised cosine
		Filter Param: 0.2
		Symbol Pt Location: NA
		Remove Q offset: NA
“CDMA2000-Base”	QPSK	Symbol rate: 1.2288 MHz
		Meas Filter: IS95 TXEQ_MEA
		Ref Filter: IS95 REF
		Filter Param: None
		Symbol Pt Location: NA
		Remove Q offset: NA
“W-CDMA”	QPSK	Symbol rate: 3.84 MHz
		Meas Filter: Root raised cosine
		Ref Filter: Raised cosine
		Filter Param: 0.22
		Symbol Pt Location: NA
		Remove Q offset: NA

In addition to the specific settings listed in the table, the following general settings are also made when you load any of the defined presets.

Setting	Value
Points/symbol	4
Burst detection mode	Off
Burst detection threshold	-10 dBc
Analysis offset	Auto
Analysis length	Auto

Preset Name	Modulation type	Settings
Frequency offset	Auto	
Measurement BW	Auto	
Frequency deviation	Auto	
Modulation index	Auto	

Examples

`SENSe:DDEMOD:SOURCE1:PRESET "802.15.4"` the measurement source1 demodulation to the standard OQPSK modulation.

[SENSe]:DDEMod[:SOURce<x>]:RING:RADius

Sets or queries the ring radius of the measurement source in the digital APSK modulation analysis.

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:RING:RADius <value>
[SENSe]:DDEMod[:SOURce<x>]:RING:RADius?
```

Arguments

`<x>` = 1 to 8, representing Range of Source.

`<value>::<NRf>` specifies the ring radius.

APSK16 valid ring arguments are 1 and 2.

APSK32 valid ring arguments are 1, 2, and 3.

Examples

`SENSe:DDEMOD:SOURce1:RING:RADIUS`

[SENSe]:DDEMod[:SOURce<x>]:RING:ROTation

Sets or queries the ring rotation of the measurement source in the digital APSK modulation analysis.

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:RING:ROTation <value>
[SENSe]:DDEMod[:SOURce<x>]:RING:ROTation?
```

Arguments

<x> = 1 to 8, representing Range of Source.

<value>::<NRf> specifies the ring rotation.

APSK16 valid ring arguments are 1 and 2.

APSK32 valid ring arguments are 1, 2, and 3.

Examples

SENSe:DDEMOD[:SOURce<x>]:RING:ROTATION

[SENSe]:DDEMod[:SOURce<x>]:SRATe

Sets or queries the symbol rate for the measurement source in the digital modulation analysis.

Conditions

Measurement views: General purpose digital modulation

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:SRATe<value>
[SENSe]:DDEMod[:SOURce<x>]:SRATe?
```

Arguments

<x> = 1 to 8, representing Range of Source.

<value>::<NRf> specifies the symbol rate.

Standard product range: 100 Hz to 40 MHz.

Option 110 product range: 100 Hz to 110 MHz.

Examples

SENSe:DDEMOD:SOURCE1:SRATE 21.0E3 sets the symbol rate of measurement source1 to 21 kHz.

[SENSe]:DDEMod[:SOURce<x>]:SWAP:IQ

Determines whether or not to exchange I and Q data of the measurement source before demodulating.

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:SWAP:IQ { OFF | ON | 0 | 1 }
```

```
[SENSe]:DDEMod[:SOURce<x>]:SWAP:IQ?
```

Arguments

<x> = 1 to 8, representing Range of Source.

`OFF` or `0` uses I and Q data as they are.

`ON` or `1` exchanges I and Q data.

Examples

`SENSe:DDEMOD:SOURce1:SWAP:IQ ON` exchanges I and Q data of the measurement source1 before demodulating.

[SENSe]:DDEMod[:SOURce<x>]:SYMBol:HSSHift

Sets or queries the measurement source Q data half-symbol shift for OQPSK and SOQPSK signals.

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:SYMBol:HSSHift { OFF | ON | 0 | 1 }
[SENSe]:DDEMod[:SOURce<x>]:SYMBol:HSSHift?
```

Related Commands

[SENSe]:DDEMod[:SOURce<x>]:MODulation:TYPE (on page 2037)

Arguments

`<x>` = 1 to 8, representing Range of Source.

`OFF` or `0` the Q offset shift is not applied.

`ON` or `1` the Q offset shift is applied.

Examples

`SENSe:DDEMod:SOURce1:SYMBol:HSSHift ON` the measurement source Q offset is applied.

[SENSe]:DDEMod[:SOURce<x>]:SYMBol:MAP:SOURce? (Query only)

Queries the user symbol map of the measurement source.

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:SYMBol:MAP:SOURce?
```

Arguments

<x> = 1 to 8, representing Range of Source.

Examples

SENSe:DDEMOD:SOURCE1:SYMBOL:MAP:SOURCE? returns the symbol map filename of the measurement source1.

[SENSe]:DDEMod[:SOURce<x>]:SYMBol:MAP[:STATe]

Determines whether or not to use the user symbol map of the measurement source.

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:SYMBol:MAP[:STATe] { OFF | ON | 0 | 1 }
[SENSe]:DDEMod[:SOURce<x>]:SYMBol:MAP[:STATe]?
```

Related Commands

[SENSe]:DDEMod[:SOURce<x>]:SYMBol:MAP:SOURce<x> (on page 2044)

Arguments

<x> = 1 to 8, representing Range of Source.

OFF or 0 disables the user symbol map.

ON or 1 enables the user symbol map.

Examples

SENSe:DDEMOD:SOURCE1:SYMBOL:MAP:STATE ON enables the user symbol map of measurement source1.

[SENSe]:DDEMod[:SOURce<x>]:SYMBol:PLOT:POSition

Sets or queries the symbol point location for measurement source on an SOQPSK waveform.

Conditions

Source keyword is optional, if not present Source1 will be default source.

Effective on SOQPSK waveform and any general purpose digital demodulation views except Signal Quality and Symbol Table.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:SYMBol:PLOT:POSition { EDGe | MIDDle }
```

```
[SENSe]:DDEMod[:SOURce<x>]:SYMBol:PLOT:POSition?
```

Arguments

<x> = 1 to 8, representing Range of Source.

EDGe sets the symbol location at the edge of the eye opening.

MIDDle sets the symbol location at the middle of the eye opening.

Examples

SENSe:DDEMOD:SOURCE1:SYMBOL:PLOT:POSITION EDGE sets the symbol location the edge of the eye opening for the measurement source 1.

[SENSe]:DDEMod[:SOURce<x>]:SYMBol:POINts

Sets or queries the number of points per symbol of the measurement source (how many points to use between symbols when connecting the dots).

NOTE: 1 is not valid for the GMSK modulation.

NOTE: In the constellation view, select VECTors using the TRACe<x>:CONSte:MODE command first to change Points/Symbol.

Conditions

Measurement views: Constellation, EVM versus Time, Magnitude error versus Time, Phase error versus Time, Freq Dev versus Time, Demod I&Q versus Time.

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:SYMBol:POINts { ONE | TWO | FOUR | EIGHt }
[SENSe]:DDEMod[:SOURce<x>]:SYMBol:POINts?
```

Arguments

<x> = 1 to 8, representing Range of Source.

ONE, TWO, FOUR, and EIGHT represent the number of points per symbol.

The following table lists the conditions of use.

Argument	Supported modulation type
ONE	All except SOQPSK, OQPSK, SBPSK
TWO	All except SOQPSK
THREE	All
FOUR	All

Examples

`SENSe:DDEMOD:SOURCE1:SYMBOL:POINTS FOUR` sets the number of points per symbol of the measurement source1 to four.

[SENSe]:DDEMod[:SOURce<x>]:SYMBol:RATE:SEARch

Determines whether to enable a symbol rate search of the measurement source. This command is valid when the demodulation type is set to FSK2, FSK4, FSK8, or FSK16.

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:SYMBol:RATE:SEARch { OFF | ON | 0 | 1 }
```

Related Commands

[SENSe]:DDEMod[:SOURce<x>]:MODulation:TYPE (on page 2037)

Arguments

`<x>` = 1 to 8, representing Range of Source.

`OFF` or `0` disables the search.

`ON` or `1` enables the search.

Examples

`SENSe:DDEMOD:SOURCE1:SYMBOL:RATE:SEARCH ON` enables the symbol rate search of measurement source1.

[SENSe]:DDEMod[:SOURce<x>]:SYNCh:WORD

Determines whether to enable the synchronization word of the measurement source.

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:SYNCh:WORD { OFF | ON | 0 | 1 }
[SENSe]:DDEMod[:SOURce<x>]:SYNCh:WORD?
```

Related Commands

[SENSe]:DDEMod[:SOURce<x>]:SYNCh:WORD:SYMBol (on page 2048)

Arguments

`<x>` = 1 to 8, representing Range of Source.

OFF or 0 disables the synchronization word.

ON or 1 enables the synchronization word.

Examples

SENSe:DDEMOD:SOURCE1:SYNCH:WORD ON enables the synchronization word of the measurement source1.

[SENSe]:DDEMod[:SOURce<x>]:SYNCh:WORD:SYMBol

Sets or queries the synchronization word of the measurement source when [SENSe]:DDEMod[:SOURce<x>]:SYNCh:WORD is ON. The word depends on the modulation type selected by the [SENSe]:DDEMod[:SOURce<x>]:MODulation:TYPE command.

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:SYNCh:WORD:SYMBol<block>
[SENSe]:DDEMod[:SOURce<x>]:SYNCh:WORD:SYMBol?
```

Arguments

<x> = 1 to 8, representing Range of Source.

<block>::=#<num_digit><num_byte><sym(1)><sym(2)>...<sym(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<sym(n)> is the n^{th} symbol value of the sync word. 32-bit integer.

n: Max 256.

Examples

SENSe:DDEMOD:SOURce1:SYNCH:WORD:SYMBOL #216xxxx (4 symbols) sets a sync word composed of four symbols for the measurement source1.

[SENSe]:DDEMod[:SOURce<x>]:TIMe:UNITs

Sets or queries the fundamental unit of time of the measurement source.

Conditions

Measurement views: General purpose digital modulation

Source keyword is optional, if not present Source 1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:DDEMod[:SOURce<x>]:TIMe:UNITs { SEConds | SYMBols }
[SENSe]:DDEMod[:SOURce<x>]:TIMe:UNITs?
```

Arguments

<x> = 1 to 8, representing Range of Source.

SEConds specifies the fundamental unit of time as seconds.

SYMBols specifies the fundamental unit of time as symbols.

Examples

SENSe:DDEMOD:SOURCE1:TIME:UNITS SEConds specifies the fundamental unit of time as seconds for measurement source 1.

[SENSe]:DIQVtime[:MEASview<y>]:SELEcted:SOURce

Sets or queries the Source of Demod I&Q versus Time for a selected Measurement view number.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Demod I&Q versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Demod I&Q versus Time).

Source cannot be changed when acquisition or analysis is in progress. Stop the acquisition and change the Source.

Group

Sense commands

Syntax

```
[SENSe]:DIQVtime[:MEASview<y>]:SELEcted:SOURce {"Source 1" "Source 2" "Source 3"| "Source 4"
[SENSe]:DIQVtime[:MEASview<y>]:SELEcted:SOURce?
```

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1.

Examples

SENSe:DIQVtime:MEASview1:SELECTED:SOURce? queries the Source for Demod I&Q versus Time Measurement View .

[SENSe]:DPX:AUDio:DEMod:GAIN

Sets or queries the audio gain.

NOTE: The sound level is also affected by the Windows volume control.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:AUDio:DEMod:GAIN <value>
[SENSe]:DPX:AUDio:DEMod:GAIN?
```

Arguments

`<value>::<NR1>` specifies the audio gain. Range: 0 to 15 (integer).

Examples

`SENSe:DPX:AUDIO:DEMOD:GAIN 7` sets the audio gain to 7.

[SENSe]:DPX:AUDio:DEMod:RXBWidth

Sets or queries the receiver bandwidth in the audio demodulation.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:AUDio:DEMod:RXBWidth <value>
[SENSe]:DPX:AUDio:DEMod:RXBWidth?
```

Arguments

`<value>::<NRf>` specifies the receiver bandwidth in the audio demodulation.

Range: 1 kHz to 500 kHz.

Examples

`SENSe:DPX:AUDIO:DEMOD:RXBWIDTH 30kHz` sets the receiver bandwidth to 30 kHz.

[SENSe]:DPX:AUDio:DEMod:RXFRequency? (Query Only)

Returns the receiver frequency in the audio demodulation. The frequency depends on the setting of the `[SENSe]:DPX:AUDio:DEMod:TUNE` command.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:AUDio:DEMod:RXFRequency?
```

Arguments

None

Returns

`<frequency>::<NRf>` is the receiver frequency in the audio demodulation.

Examples

SENSe:DPX:AUDIO:DEMOD:RXFREQUENCY? might return 80.3E+6, indicating that the receiver frequency is 80.3 MHz.

[SENSe]:DPX:AUDio:DEMod:STATe

Determines whether to enable the audio demodulation.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:AUDio:DEMod:STATe { OFF | ON | 0 | 1 }
[SENSe]:DPX:AUDio:DEMod:STATe?
```

Arguments

OFF or 0 disables the audio demodulation.

ON or 1 enables the audio demodulation.

Examples

SENSe:DPX:AUDIO:DEMOD:STATE ON enables the audio demodulation.

[SENSe]:DPX:AUDio:DEMod:TUNE

Sets or queries how to determine the tuning frequency in the audio demodulation.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:AUDio:DEMod:TUNE { MR | MARK1 | MARK2 | MARK3 | MARK4 | SMARker | FREQcontrol }
[SENSe]:DPX:AUDio:DEMod:TUNE?
```

Arguments

The following table lists the arguments.

Frequency tuning

Argument	Tune with
MR	Reference marker (MR)
MARK1	Marker 1 (M1)
MARK2	Marker 2 (M2)

Argument	Tune with
MARK3	Marker 3 (M3)
MARK4	Marker 4 (M4)
SMARker	Selected marker
FREQcontrol	Center frequency setting

Examples

`SENSe:DPX:AUDIO:DEMOD:TUNE MARK1` sets the tuning frequency to the value at Marker 1.

[SENSe]:DPX:AUDio:DEMod:TYPE

Sets or queries the modulation type in the audio demodulation.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:AUDio:DEMod:TYPE { AM | FM }
[SENSe]:DPX:AUDio:DEMod:TYPE?
```

Arguments

`AM` selects the AM (Amplitude Modulation).

`FM` selects the FM (Frequency Modulation).

Examples

`SENSe:DPX:AUDIO:DEMOD:TYPE FM` selects FM in the audio demodulation.

[SENSe]:DPX:{BANDwidth|BWIDth}:ACTual? (Query Only)

Queries the actual bandwidth whether set automatically or manually.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:{BANDwidth|BWIDth}:ACTual?
```

Arguments

None

Examples

SENSe:DPX:BANDWIDTH:ACTUAL? returns the RBW as a percentage of the Span.

[SENSe]:DPX:{BANDwidth|BWIDth}:RESolution

Sets or queries the resolution bandwidth (RBW) in the DPX, Spectrum measurement.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:{BANDwidth|BWIDth}:RESolution <value>
[SENSe]:DPX:{BANDwidth|BWIDth}:RESolution?
```

Arguments

<value>::<NRf> specifies the resolution bandwidth.

Range: 1 Hz to 10 MHz.

Examples

SENSe:DPX:BANDWIDTH:RESOLUTION? might return 100.00003E+3 for the RBW.

[SENSe]:DPX:{BANDwidth|BWIDth}[:RESolution]:AUTO

Determines whether to set the resolution bandwidth (RBW) automatically or manually in the DPX, Spectrum measurement.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:{BANDwidth|BWIDth}[:RESolution]:AUTO { OFF | ON | 0 | 1 }
[SENSe]:DPX:{BANDwidth|BWIDth}[:RESolution]:AUTO?
```

Arguments

OFF or 0 specifies that the resolution bandwidth is set manually using the [SENSe]:DPX:{BANDwidth|BWIDth}:ACTual? command.

ON or 1 specifies that the resolution bandwidth is set automatically.

Examples

SENSe:DPX:BANDWIDTH:AUTO ON sets the resolution bandwidth automatically.

[SENSe]:DPX:CLEar:RESults (No Query Form)

Restarts multi-trace functions (Average and Max/Min Hold).

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:CLEar:RESults
```

Arguments

None

Examples

`SENSe:DPX:CLEAR:RESULTS` restarts multi-trace functions.

[SENSe]:DPX:COLor

Sets or queries the color palette of three-dimensional graphs.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:COLor {RED | GREen | BLUe | CYAN | BCYan | YELLow | MAGenta | GRAY | TEMPerature |
   SPECtral}
[SENSe]:DPX:COLor?
```

Arguments

The following table lists the arguments.

Color palette for DPX, Spectrum

Argument	Palette
RED	Red
GREen	Green
BLUe	Blue
CYAN	Cyan
BCYan	Binary cyan
YELLow	Yellow
MAGenta	Magenta
GRAY	Gray

Argument	Palette
TEMPerature	Temperature
SPECtral	Spectral

Examples

`SENSe:DPX:COLOR TEMPERATURE` selects the temperature color palette.

[SENSe]:DPX:COLor:MAXimum

Sets or queries the maximum value of the color axis in the DPX, Spectrum measurement.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:COLor:MAXimum <value>
[SENSe]:DPX:COLor:MAXimum?
```

Arguments

`<value> ::= <NRf>` specifies the maximum value of the color axis.

Range: The minimum value to 100%.

The minimum value is set using the `[SENSe]:DPX:COLor:MINimum` command.

Examples

`SENSe:DPX:COLOR:MAXIMUM 90` sets the maximum value of the color axis to 90%.

[SENSe]:DPX:COLor:MINimum

Sets or queries the minimum value of the color axis in the DPX, Spectrum measurement.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:COLor:MINimum <value>
[SENSe]:DPX:COLor:MINimum?
```

Arguments

`<value>::<NRf>` specifies the minimum value of the color axis.

Range: 0% to the maximum value.

The maximum value is set using the `[SENSe]:DPX:COLor:MAXimum` command.

Examples

`SENSe:DPX:COLOR:MINIMUM 10` sets the minimum value of the color axis to 10%.

[SENSe]:DPX:COLor:REVerse[:STATe]

Determines whether to reverse the color palette in the DPX, Spectrum view. When reversed, the highest density takes the color normally used for the lowest.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

`[SENSe]:DPX:COLor:REVerse[:STATe] {OFF | ON | 0 | 1}`

`[SENSe]:DPX:COLor:REVerse[:STATe]?`

Arguments

`OFF` or `0` uses the color palette in its normal order.

`ON` or `1` reverses the color palette.

Examples

`SENSE:DPX:COLOR:REVERSE:STATE ON` reverses the color palette.

[SENSe]:DPX:DDENsity:MEASurement:SHOW

Sets the Show measurement value (measurement always visible for DPX Trigger) or queries for the current value.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:DDENsity:MEASurement:SHOW { OFF | ON | 0 | 1 }
[SENSe]:DPX:DDENsity:MEASurement:SHOW?
```

Arguments

`OFF` or `0` specifies that the Show measurement value is disabled.

`ON` or `1` specifies that the Show measurement value is disabled.

Examples

`SENSe:DPX:DDENSITY:MEASUREMENT:SHOW ON` enables the Show measurement value.

SENSe:DPX:DGRam:COLor

Sets or queries the color palette of three-dimensional graphs.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:DGRam:COLor { RED | GREen | BLUe | CYAN | BCYan | YELLow | MAGenta | GRAY | TEMPerature
    | SPECtral }
[SENSe]:DPX:COLor?
```

Arguments

The following table lists the arguments.

Argument	Palette
RED	Red
GREen	Green
BLUe	Blue
CYAN	Cyan
BCYan	Binary cyan
YELLow	Yellow
MAGenta	Magenta
GRAY	Gray
TEMPerature	Temperature
SPECtral	Spectral

Examples

`SENSe:DPX:DGRam:COLOR TEMPerature` selects the temperature color palette.

[SENSe]:DPX:DGRam:COLor:MAXimum

Sets or queries the maximum value of the color axis in the DPXogram measurement.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:DGRam:COLor:MAXimum <value>
[SENSe]:DPX:DGRam:COLor:MAXimum?
```

Arguments

`<value> ::= <NRf>` specifies the maximum value of the color axis.

Range: –50 dBm to –169.00 dBm.

The minimum value is set using the `[SENSe]:DPX:DGRam:COLor:MINimum` command.

Examples

`SENSe:DPX:DGRAM:COLOR:MAXIMUM 0` sets the maximum value of the color axis to 0.00 dBm.

[SENSe]:DPX:DGRam:COLor:MINimum

Sets or queries the minimum value of the color axis in the DPXogram measurement.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:DGRam:COLor:MINimum <value>
[SENSe]:DPX:DGRam:COLor:MINimum?
```

Arguments

`<value> ::= <NRf>` specifies the maximum value of the color axis.

Range: 49.00 dBm to –170.00 dBm.

The maximum value is set using the `[SENSe]:DPX:DGRam:COLor:MAXimum` command.

Examples

`SENSe:DPX:DGRAM:COLOR:MINIMUM -100` sets the minimum value of the color axis to –100.00 dBm.

[SENSe]:DPX:DGRam:COLor:REVerse

Sets or queries reverse color scheme state.

Conditions

Applicable measurement view must be active.

Group

Sense commands

Syntax

```
[SENSe]:DPX:DGRam:COLor:REVerse {OFF | ON | 0 | 1}
[SENSe]:DPX:DGRam:COLor:REVerse?
```

Related Commands

SENSe:DPX:DGRam:COLor (on page 2056)

Arguments

`OFF` or `0` disables Reverse color scheme.

`ON` or `1` enables Reverse color scheme

Returns

`0` disables Reverse color scheme.

`1` enables Reverse color scheme.

Examples

`SENSe:DPX:DGRam:COLor:REVerse OFF` disables reverse color scheme.

[SENSe]:DPX:DGRam:POINts:COUNt

Sets or queries the number of trace points acquired for the DPXogram display.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:DGRam:POINts:COUNt { P801 | P2401 | P4001 }
[SENSe]:DPX:DGRam:POINts:COUNt?
```

Arguments

`P801` sets the number of sample points to 801.

`P2401` sets the number of sample points to 2401.

`P4001` sets the number of sample points to 4001.

Examples

`SENSe:DPX:DGRAM:POINTS:COUNT P2401` sets the number of trace points to acquire for the DPXogram display.

[SENSe]:DPX:DGRam:TIME:CAPacity? (Query Only)

Returns the time capacity of the DPXogram measurement.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:DGRam:TIME:CAPacity?
```

Arguments

None

Returns

<Time> ::= <String> The time capacity in hours:minutes:seconds.

Examples

SENSe:DPX:DGRAM:TIME:CAPACITY? might return "0:25:50" indicating that the time capacity of the measurement was 0 hours, 25 minutes, and 50 seconds.

[SENSe]:DPX:DGRam:TIME:RESolution

Sets or queries the time resolution of the DPXogram measurement.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:DGRam:TIME:RESolution <value>
[SENSe]:DPX:DGRam:TIME:RESolution?
```

Arguments

<value> ::= <NRf> specifies the time resolution of the DPXogram measurement.

Examples

SENSe:DPX:DGRAM:TIME:RESOLUTION 25.0 ms sets the time resolution to 25 ms.

[SENSe]:DPX:DGRam:TIME:RESolution:AUTO

Determines whether to set the time resolution automatically or manually in the DPXogram measurement.

When Auto is enabled, the time resolution value is based on the Time/div setting. See the DISPlay:DPX:DGRam:TIME[:SCALe]:PDIVision command.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:DGRam:TIME:RESolution:AUTO { OFF | ON | 0 | 1 }
[SENSe]:DPX:DGRam:TIME:RESolution:AUTO?
```

Arguments

OFF or 0 disables the auto time resolution.

ON or 1 enables the auto time resolution.

Examples

SENSe:DPX:DGRAM:TIME:RESOLUTION:AUTO ON sets the timer resolution automatically.

[SENSe]:DPX:DGRam:WATerfall:DIRection

Sets or queries the direction of the Waterfall (DPXogram) display.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:DGRam:WATerfall:DIRection { NW | NE }
[SENSe]:DPX:DGRam:WATerfall:DIRection?
```

Arguments

`NW` sets the display direction to northwest.

`NE` sets the display direction to northeast.

Examples

`SENSe:DPX:DGRAM:WATERFALL:DIRECTION NW` sets the waterfall display direction to northwest.

[SENSe]:DPX:DGRam:WATerfall:ENABle

Sets or queries the Waterfall (DPXogram) display.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:DGRam:WATerfall:ENABle { OFF | ON | 0 | 1 }
[SENSe]:DPX:DGRam:WATerfall:ENABle?
```

Arguments

`OFF` or `0` disables the Waterfall display.

`ON` or `1` enables the Waterfall display.

Examples

`SENSe:DPX:DGRAM:WATERFALL:ENABLE ON` enables the 3-D Waterfall display.

[SENSe]:DPX:DGRam:Y[:SCALe]:AUTO (No Query Form)

Rescales the height axis automatically to fit the waveform to the screen in the DPXogram display.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:DGRam:Y[:SCALe]:AUTO
```

Arguments

None

Examples

`SENSe:DPX:DGRAM:Y[:SCALE]:AUTO` rescales the height scale automatically to fit the waveform to the DPXogram display.

[SENSe]:DPX:DGRam:Y[:SCALe]:RESet (No Query Form)

Resets the height scale of the DPXogram display to the default values:

Height position = 0 dB and Height scale = 100 dB.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:DGRam:Y[:SCALe]:RESet
```

Arguments

None

Examples

`SENSe:DPX:DGRAM:Y[:SCALE]:RESET` resets the height scale of the DPXogram display.

[SENSe]:DPX:DWELl

Sets or queries the value of the Dwell time for the DPX, Spectrum measurement.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:DWELl <value>
[SENSe]:DPX:DWELl?
```

Arguments

`<value>::<NRf>` specifies the time the DPX sweep remains in a frequency segment collecting data and updating the bitmap and traces before moving on to the next higher frequency segment.

Range: 50 ms to 100 s per frequency segment.

Examples

`SENSe:DPX:DWELl 100ms` sets the dwell value for sweeps to 100 ms.

[SENSe]:DPX:DWELI:AUTO

Sets the value of the Dwell time automatically or queries for the current value.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:DWELl:AUTO { OFF | ON | 0 | 1 }
[SENSe]:DPX:DWELl:AUTO?
```

Related Commands

[SENSe]:DPX:DWELI (on page 2062)

Arguments

`OFF` or `0` specifies that the dwell is set manually.

`ON` or `1` specifies that the dwell is set automatically.

Examples

`SENSe:DPX:DWELl:AUTO ON` sets the dwell value automatically for sweeps.

[SENSe]:DPX:FREQuency:CENTer

Sets or queries the center frequency in the DPX, Spectrum measurement.

NOTE: The center, start and stop frequencies are set interlocking each other with the following relationships: (start frequency) = (center frequency) - (span)/2 and (stop frequency) = (center frequency) + (span)/2.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:FREQuency:CENTer <value>
[SENSe]:DPX:FREQuency:CENTer?
```

Related Commands

[SENSe]:DPX:FREQuency:STARt (on page 2065)

[SENSe]:DPX:FREQuency:STOP (on page 2066)

Arguments

`<value> ::= <NRf>` specifies the center frequency.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples

`SENSe:DPX:FREQUENCY:CENTER 7.5GHz` sets the center frequency to 7.5 GHz.

[SENSe]:DPX:FREQuency:SPAN

Sets or queries the frequency span in the DPX, Spectrum measurement.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:FREQuency:SPAN <value>
[SENSe]:DPX:FREQuency:SPAN?
```

Arguments

`<value> ::= <NRf>` is the frequency span.

Range: 10 Hz to 40 MHz (Standard) / 110 MHz (Option 110)

Examples

`SENSe:DPX:FREQUENCY:SPAN 20MHz` sets the span to 20 MHz.

[SENSe]:DPX:FREQuency:SPAN:{BANDwidth|BWIDth} [:RESolution]:RATio

Determines whether to set the resolution bandwidth (RBW) automatically or manually in the DPX, Spectrum measurement.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:FREQuency:SPAN:{BANDwidth|BWIDth}[:RESolution]:RATio <value>
[SENSe]:DPX:FREQuency:SPAN:{BANDwidth|BWIDth}[:RESolution]:RATio?
```

Related Commands

[SENSe]:DPX:{BANDwidth|BWIDth}[:RESolution]:AUTO (on page 2053)

Arguments

`<value> ::= <NRf>` is used to calculate the RBW.

Examples

`SENSe:DPX:FREQuency:SPAN:BANDWIDTH:RATIO 100` the ratio of 100 is used to set the resolution bandwidth when `SENSe:DPX:{BANDwidth|BWIDth}[:RESolution]:AUTO` is set to ON.

[SENSe]:DPX:FREQuency:STARt

Sets or queries the measurement start frequency (left edge on the graph) in the DPX, Spectrum measurement.

The center, start and stop frequencies are set interlocking each other. Refer to the `[SENSe]:DPX:FREQuency:CENTer` command.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:FREQuency:STARt <value>
[SENSe]:DPX:FREQuency:STARt?
```

Related Commands

[SENSe]:DPX:FREQuency:STOP (on page 2066)

Arguments

`<value> ::= <NRf>` is the measurement start frequency.

Range: (center frequency) ± (span)/2.

Examples

`SENSe:DPX:FREQUENCY:START 6.95GHz` sets the start frequency to 6.95 GHz.

[SENSe]:DPX:FREQuency:STEP

Sets or queries the frequency step size (the amount per press by which the up or down key changes the setting value). Programming a specified step size sets `[SENSe]:DPX:FREQuency:STEP:AUTO OFF`.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:FREQuency:STEP <value>
[SENSe]:DPX:FREQuency:STEP?
```

Related Commands

[SENSe]:DPX:FREQuency:STEP:AUTO (on page 2066)

Arguments

`<value> ::= <NRf>` specifies the frequency step size.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples

`SENSE:DPX:FREQUENCY:STEP 1.5kHz` sets the step size to 1.5 kHz.

[SENSe]:DPX:FREQuency:STEP:AUTO

Determines whether to set the frequency step size automatically or manually.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:FREQuency:STEP:AUTO { OFF | ON | 0 | 1 }
[SENSe]:DPX:FREQuency:STEP:AUTO?
```

Arguments

`OFF` or `0` specifies that the frequency step size is set manually using the `[SENSe]:DPX:FREQuency:STEP` command.

`ON` or `1` specifies that the frequency step size is set automatically.

Examples

`SENSE:DPX:BANDWIDTH:AUTO ON` sets the frequency step size automatically.

[SENSe]:DPX:FREQuency:STOP

Sets or queries the measurement stop frequency (right edge of the graph) in the DPX, Spectrum measurement.

The center, start and stop frequencies are set interlocking each other. Refer to the `[SENSe]:DPX:FREQuency:CENTer` command.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:FREQuency:STOP <value>
[SENSe]:DPX:FREQuency:STOP?
```

Related Commands

[SENSe]:DPX:FREQuency:STARt (on page 2065)

Arguments

<value>::<NRf> is the measurement stop frequency.

Range: (center frequency) ± (span)/2.

Examples

SENSe:DPX:FREQUENCY:STOP 7.05GHz sets the stop frequency to 7.05 GHz.

[SENSe]:DPX:PLOT

Sets or queries the plot in the DPX, Spectrum measurement.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:PLOT { DGRam | SPLit | SPECtrum | ZSPan | FREQuency | PHASe }
[SENSe]:DPX:PLOT?
```

Arguments

DGRam selects the DPXogram display.

SPLit selects the DPX Split display.

SPECtrum selects the DPX, Spectrum display.

ZSPan select the DPX Zero Span display.

FREQuency selects the DPX Frequency vs Time display.

PHASe selects the DPX Phase vs Time display.

Examples

SENSe:DPX:PLOT SPECTRUM selects the DPX, Spectrum display

[SENSe]:DPX:POINts:COUNt

Sets the number of trace points acquired for the DPX, Spectrum display.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:POINts:COUNt { P801 | P2401 | P4001 | P10401 }
[SENSe]:DPX:POINts:COUNt?
```

Arguments

P801 sets the number of sample points to 801.

P2401 sets the number of sample points to 2401.

P4001 sets the number of sample points to 4001.

P8001 sets the number of sample points to 8001.

P10401 sets the number of sample points to 10401.

Examples

SENSe:DPX:POINTS:COUNT P2401 sets the number of trace points to acquire for the DPX, Spectrum display.

[SENSe]:DPX:TDM:{BANDwidth|BWIDth}

Sets or queries the value of the measurement bandwidth in the DPX measurement.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:TDM:{BANDwidth|BWIDth} <value>
[SENSe]:DPX:TDM:{BANDwidth|BWIDth}?
```

Arguments

<value>::<NRf> specifies the measurement bandwidth.

Range: 100 Hz to Max AcqBW.

Examples

SENSe:DPX:TDM:BANDWIDTH 200 sets the measurement bandwidth to 200 Hz.

[SENSe]:DPX:TDM:{BANDwidth|BWIDth}:ACTual? (Query Only)

Queries the actual bandwidth in the DPX measurement.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:TDM:{BANDwidth|BWIDth}:ACTual?
```

Arguments

None

Returns

<NRf> The actual RBW in Hz.

Examples

SENSe:DPX:TDM:BANDWIDTH:ACTUAL? might return 10E+3, indicating that the actual bandwidth is 10 kHz

[SENSe]:DPX:TDM:FREQuency:CENTer

Sets or queries the center frequency in the DPX, Spectrum measurement.

NOTE: The center, start and stop frequencies are set interlocking each other with the following relationships: (start frequency) = (center frequency) - (span)/2 and (stop frequency) = (center frequency) + (span)/2.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:TDM:FREQuency:CENTer <value>
[SENSe]:DPX:TDM:FREQuency:CENTer?
```

Arguments

<value>::<NRf> specifies the center frequency.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples

SENSe:DPX:TDM:FREQUENCY:CENTER 7.5GHz sets the center frequency to 7.5 GHz.

[SENSe]:DPX:TDM:FREQuency:TXBer:CLEar:RESults (No Query Form)

Clears the measurement results of the active Tx BER measurement. Sending this command does not halt the Tx BER measurement.

Conditions

Measurement views: DPX, frequency

Group

Sense commands

Syntax

```
[SENSe]:DPX:TDM:FREQuency:TXBer:CLEar:RESults
```

Arguments

None

Examples

`SENSe:DPX:TDM:FREQUENCY:TXBER:CLEAR:RESULTS` clears the results of the active TxBER measurement, but does not halt the measurement.

[SENSe]:DPX:TDM:FREQuency:TXBer:ENABle

Sets or queries the Tx BER measurement.

Conditions

Measurement views: DPX, frequency

Group

Sense commands

Syntax

```
[SENSe]:DPX:TDM:FREQuency:TXBer:ENABle { OFF | ON | 0 | 1 }
```

Arguments

`OFF` or `0` halts the Tx BER measurement.

`ON` or `1` starts the Tx BER measurement.

Returns

Examples

```
SENSe:DPX:TDM:FREQUENCY:TXBER:ENABLE
```

[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern

Sets or queries the Tx BER data pattern.

Conditions

Measurement views: DPX frequency

Group

Sense commands

Syntax

```
[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern { PRBS9 | PRBS11 | PRBS15 | USER }
[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern?
```

Arguments

`PRBS9` specifies a 9-bit PRBS as the data pattern.

`PRBS11` specifies a 11-bit PRBS as the data pattern.

`PRBS15` specifies a 15-bit PRBS as the data pattern

`USER` specifies a user-defined sequence as the data pattern.

Returns

`<string>` which is the selected data pattern. Returned strings are the same as the arguments.

Examples

`SENSe:DPX:TDM:FREQUENCY:TXBER:PATTERN PRBS9` sets the data pattern to PRBS9.

[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern:INVert

Sets or queries the data pattern Invert setting.

Conditions

Measurement views: DPX frequency

Group

Sense commands

Syntax

```
[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern:INVert { 0 | 1 | OFF | ON }
[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern:INVert?
```

Arguments

`OFF` or `0` sets the data pattern to normal.

`ON` or `1` inverts the data pattern.

Returns

`<value> ::= <NR1>` where:

`0` indicates the data pattern is normal.

`1` indicates the data pattern is inverted.

Examples

`SENSe:DPX:TDM:FREQUENCY:TXBER:PATTERN:INVERT 1` inverts the data pattern.

[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern:REPeat

Sets or queries the test duration for a User Pattern. A User Pattern test duration can be one time through or it can repeat.

Conditions

Measurement views: DPX frequency

Group

Sense commands

Syntax

```
[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern:REPeat { 0 | 1 | OFF | ON }
[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern:REPeat?
```

Arguments

`OFF` or `0` sets the data pattern to normal.

`ON` or `1` inverts the data pattern.

Returns

<value> ::= <NR1> where:

0 indicates repeat is off.

1 indicates repeat is on.

Examples

SENSe:DPX:TDM:FREQUENCY:TXBER:PATTERN:REPEAT 1 causes the User pattern to repeat.

[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern:REVerse

Sets or queries whether the data pattern is in normal order or reversed. This setting applies only to PRBS patterns.

Conditions

Measurement views: DPX frequency

Group

Sense commands

Syntax

```
[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern:REVerse { 0 | 1 | OFF | ON }
[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern:REVerse?
```

Arguments

OFF or 0 sets the data pattern to normal order.

ON or 1 sets the data pattern to reversed order.

Returns

<value> ::= <NR1> where:

0 indicates the data pattern is in normal order.

1 indicates the data pattern is in reversed order.

Examples

SENSe:DPX:TDM:FREQUENCY:TXBER:PATTERN:REVERSE? might return 1, indicating the data pattern is reversed.

[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern:SOURce? (Query Only)

Returns the path and file name of the user-defined pattern file.

Conditions

Measurement views: DPX frequency

Group

Sense commands

Syntax

```
[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern:SOURce?
```

Returns

<path>::<string> which is the path to the defined User data pattern.

Examples

SENSe:DPX:TDM:FREQUENCY:TXBER:PATTERN:SOURCE? might return "C:\RSA5100B Files\Example Files\SampleUserPattern.csv".

[SENSe]:DPX:TDM:RBW

Sets or queries the value of the RBW filter.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:TDM:RBW <value>
[SENSe]:DPX:TDM:RBW?
```

Arguments

<value>::<NRf> specifies the RBW filter value.

Range: 1 Hz to 60MHz

Examples

SENSe:DPX:TDM:RBW 2.5 sets the RBW filter value to 2.5 Hz.

[SENSe]:DPX:TDM:RBW:STATe

Determines whether to set the resolution bandwidth (RBW) filter automatically or manually in the DPX measurement.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:TDM:RBW:STATe { OFF | ON | 0 | 1 }
[SENSe]:DPX:TDM:RBW:STATe?
```

Arguments

OFF or 0 specifies that the RBW is set automatically.

ON or 1 specifies that the RBW is set manually.

Examples

SENSe:DPX:TDM:RBW:STATE ON sets the RBW manually.

[SENSe]:DPX:TDM:SWEep:TIME

Sets or queries the sweep time when the plot is set to ZSPan, FREQuency, or PHASe.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:TDM:SWEep:TIME <value>
[SENSe]:DPX:TDM:SWEep:TIME?
```

Arguments

`<value>::<NRf>` specifies the sweep time.

Range: 100 ηs to 2000 s.

Examples

`SENSe:DPX:TDM:SWEEP:TIME 200` sets the sweep time to 200 s.

[SENSe]:DPX:TDM:TMOTion

Sets or queries the trace motion when the plot is set to ZSPan, FREQuency, or PHASe.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:TDM:TMOTion { NONE | ROLL | NORMal }
[SENSe]:DPX:TDM:TMOTion?
```

Arguments

`NONE` turns scrolling off.

`ROLL` shifts the previous data points to the left and the latest data point to the right edge of the display.

`NORMAL` shifts the latest data point on the display from left to right.

Examples

`SENSe:DPX:TDM:TMOTION NORMAL` sets the scroll direction to normal.

[SENSe]:EDIagram[:MEASview<y>]:SELEcted:SOURce

Sets or queries the Source of Eye diagram for a selected Measurement view number.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Eye diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Eye diagram).

Source cannot be changed when acquisition or analysis is in progress. Stop the acquisition and change the Source.

Group

Sense commands

Syntax

```
[SENSe]:EDIagram[:MEASview<y>]:SELEcted:SOURce {"Source 1" | "Source 2" | "Source 3" | "Source 4"}
[SENSe]:EDIagram[:MEASview<y>]:SELEcted:SOURce?
```

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1.

Examples

`SENSe:EDIagram:MEASview1:SELEcted:SOURce?` queries the Source for Eye diagram Measurement View 1.

[SENSe]:DEMOd:EVM[:MEASview<y>]:SELEcted:SOURce

Sets or queries the Source of EVM versus Time for a selected Measurement view number.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Eye diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Eye diagram).

Source cannot be changed when acquisition or analysis is in progress. Stop the acquisition and change the Source.

Group

Sense commands

Syntax

```
[SENSe]:DEMOd:EVM[:MEASview<y>]:SELEcted:SOURce {"Source 1" | "Source 2" | "Source 3" | "Source 4"}
[SENSe]:DEMOd:EVM[:MEASview<y>]:SELEcted:SOURce?
```

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1.

Examples

`SENSe:DEMOd:EVM:MEASview1:SELEcted:SOURce?` queries the Source for EVM versus Time Measurement View 1.

SENSe:EMC:ACCEssory<x>:ANTEnna:DISTance

Sets or queries the distance between EUT and Antenna in EMC View.

<x> = <NR1> 1 to 6

Conditions

Measurement views: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:ACCEssory<x>:ANTEnna:DISTance <value>
SENSe:EMC:ACCEssory<x>:ANTEnna:DISTance?
```

Arguments

`<value> = <NR3>`

Returns

<NR3>

Examples

`SENSe:EMC:ACCEssory1:ANTEnna:DISTance` might returns `10.5`, indicating that the distance between accessory 1 (Antenna) and EUT is 10.5 m.

SENSe:EMC:ACCEssory<x>:ASSEt:TAG

Sets or queries the asset tag of the specified accessory in EMC View.

<x> = <NR1> 1 to 6

Conditions

Measurement views: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:ACCEssory<x>:ASSEt:TAG <arg>
SENSe:EMC:ACCEssory<x>:ASSEt:TAG?
```

Arguments

`<arg> = "String"`

Returns

<String>

Examples

`SENSe:EMC:ACCEssory1:ASSEt:TAG` might return `A4001993`, indicating that asset tag of accessory 1 is A4001993.

SENSe:EMC:ACCEssory<x>:CALibration:DATE

Sets or queries the Calibration date of the specified accessory in EMC View.

<x> = <NR1> 1 to 6

Conditions

Measurement views: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:ACCEssory<x>:CALibration:DATE <arg>
SENSe:EMC:ACCEssory<x>:CALibration:DATE?
```

Arguments

`<arg> = "String"`

Returns

<String>

Examples

`SENSe:EMC:ACCEssory1:CALibration:DATE` might return `"9/21/2017"`, indicating that calibration date of the accessory 1 is 9/21/2017 (21st of September 2017).

SENSe:EMC:ACCEssory<x>:CALibration:DATE:DISAble

Sets or queries the enabling and disabling the calibration due date of the specified accessory in EMC View.

<x> = <NR1> 1 to 6

Conditions

Measurement views: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:ACCEssory<x>:CALibration:DATE:DISAble { 0 | OFF | 1 | ON}
SENSe:EMC:ACCEssory<x>:CALibration:DATE:DISAble?
```

Arguments

`0` or `OFF` disables the calibration due date check box

`1` or `ON` enables the calibration due date check box

Returns

<Boolean>0,1

Examples

`SENSe:EMC:ACCEssory1:CALibration:DATE:DISAble 0` will disable the calibration due date check box of accessory 1.

SENSe:EMC:ACCEssory<x>:CONStant:LOSS

Sets or queries the Constant loss of an accessory in EMC View.

<x> = <NR1> 1 to 6

Conditions

Measurement views: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:ACCEssory<x>:CONStant:LOSS <value>
SENSe:EMC:ACCEssory<x>:CONStant:LOSS?
```

Arguments

`<value> = <NR3>`

Returns

<NR3>

Examples

`SENSe:EMC:ACCEssory1:CONStant:LOSS` might return `10.00`, indicating that the Constant Loss of accessory 1 is 10 dB.

SENSe:EMC:ACCEssory<x>:ENABle

Sets or queries enabling and disabling the specified accessory in EMC View.

<x> = <NR1> 1 to 6

Conditions

Measurement views: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:ACCEssory<x>:ENABle { 0 | OFF | 1 | ON}
SENSe:EMC:ACCEssory<x>:ENABle?
```

Arguments

`0` or `OFF` disables the accessory

`1` or `ON` enables the accessory

Returns

<Boolean>0,1

Examples

`SENSe:EMC:ACCEssory<x>:ENABle 0` will disable accessory 1.

SENSe:EMC:ACCEssory<x>:FREQuency:INTerpolation

Sets or queries the frequency interpolation of an accessory in EMC View.

`<x> = <NR1> 1 to 6`

Conditions

Measurement views: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:ACCEssory<x>:FREQuency:INTerpolation { LIN | LOG}
SENSe:EMC:ACCEssory<x>:FREQuency:INTerpolation?
```

Arguments

`LIN` = Sets Linear as frequency interpolation

`LOG` = Sets Log as frequency interpolation

Returns

<String> LIN or LOG

Examples

`SENSe:EMC:ACCEssory1:ANTEnna:DISTance` might returns `10.5`, indicating that the distance between accessory 1 (Antenna) and EUT is 10.5 m.

SENSe:EMC:ACCEssory<x>:LOSS:TABLe:ADD (No Query Form)

Adds a row in the loss table with values for the specified accessory in EMC view.

<x> = <NR1> 1 to 6

NOTE: This command works only when the specified accessory is enabled.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:ACCEssory<x>:LOSS:TABLe:ADD <arg1,arg2>
```

Arguments

`<arg1>` = Specifies the frequency value

`<arg2>` = Specifies the Gain/Loss value

Returns

None

Examples

`SENSe:EMC:ACCEssory1:LOSS:TABLe:ADD 30KHz ,10` adds a new row to the accessory 1 loss table with frequency 30 kHz and an RBW of 10 k.

SENSe:EMC:ACCEssory<x>:LOSS:TABLe:DELete:ALL (No Query Form)

Deletes all the row in the loss table for the specified accessory in EMC view.

<x> = <NR1> 1 to 6

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:ACCEssory<x>:LOSS:TABLe:DELete:ALL
```

Arguments

None

Returns

None

Examples

`SENSe:EMC:ACCEssory2:LOSS:TABLe:DELete:ALL` deletes all the rows in the loss table of accessory 2.

SENSe:EMC:ACCEssory<x>:LOSS:TYPE

Sets or queries the loss type of an accessory in EMC View.

<x> = <NR1> 1 to 6

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:ACCEssory<x>:LOSS:TYPE {ALT | CLOS | AFT}
SENSe:EMC:ACCEssory<x>:LOSS:TYPE?
```

Arguments

`ALT` = Antenna Loss Table

`CLOS` = Constant Loss

`AFT` = Antenna Factor

Returns

`<String>` ALT or CLOS or AFT

Examples

`SENSe:EMC:ACCEssory1:LOSS:TYPE?` might return `CLOS`, indicating that the loss type of accessory 1 is CLOS (Constant Loss).

SENSe:EMC:ACCEssory<x>:MODel

Sets or queries the model number of the specified accessory in EMC View.

<x> = <NR1> 1 to 6

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:ACCEssory<x>:MODel <arg>
SENSe:EMC:ACCEssory<x>:MODel?
```

Arguments

`<arg> = "String"`

Returns

<String>

Examples

`SENSe:EMC:ACCEssory2:MODel?` might return `A16235`, indicating that the model number of accessory 2 is A16235.

SENSe:EMC:ACCEssory<x>:SERial

Sets or queries the serial number of the specified accessory in EMC View.

<x> = <NR1> 1 to 6

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:ACCEssory<x>:SERial <arg>
SENSe:EMC:ACCEssory<x>:SERial?
```

Arguments

`<arg> = "String"`

Returns

<String>

Examples

SENSe:EMC:ACCEssory1:SERial might return S1987652, indicating that the serial number of accessory 1 is S1987652.

SENSe:EMC:ACCEssory<x>:TYPE

Sets or queries the type of specified accessory in EMC view.

<x> = <NR1> 1 to 6

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:ACCEssory<x>:TYPE <arg>
SENSe:EMC:ACCEssory<x>:TYPE?
```

Arguments

If emission type is Conducted:

If emission type is Radiated:

<arg> = {ANTE | NFPR | CABL | AMPL | LIMI | ATTE | FILT | OTHE }

where

LISN = LISN

CABL = Cable

LIMI = Limiter

AMPL = Amplifier

ATTE = Attenuator

FILT = Filter

OTHE = Others

ANTE = Antenna

NFPR = Near Field Probe

Returns

<String> See Arguments

Examples

SENSe:EMC:ACCEssory1:TYPE might return LIMI, indicating that accessory 1 is Limiter.

SENSe:EMC:CLEar:RESults (No Query Form)

Clears all the data in EMC view.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:CLEar:RESults
```

Arguments

None

Returns

None

Examples

`SENSE:EMC:CLEAR:RESULTS` clears the data in the EMC-EMI display.

SENSe:EMC:EMISsion:TYPE

Sets or queries the type of emission in EMC View.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:EMISsion:TYPE {COND | RADI}
SENSe:EMC:EMISsion:TYPE?
```

Arguments

`COND` = Sets Conducted as Emission type

`RADI` = Sets Radiated as Emission type

Returns

<String> COND or RADI

Examples

`SENSE:EMC:EMISSION:TYPE` sets Conducted as the emission type.

SENSe:EMC[:FREQuency]:OVERlap? (Query Only)

Queries whether there is a frequency range overlap in ranges and Limits of EMC view.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC[:FREQuency]:OVERlap?
```

Arguments

None

Returns

<Boolean> 0,1

Examples

`SENSE:EMC[:FREQUENCY]:OVERLAP?` might return `1`, indicating that the frequency is overlapped.

SENSe:EMC:LIST

Sets or queries the list of spots to be displayed in EMC View.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:LIST {ALL | OVER | OLWM}
SENSe:EMC:LIST?
```

Arguments

`ALL` = Sets All spots to be displayed

`OVER` = Sets Over Limit spots to be displayed

`OLWM` = Sets Over Limit with margin spots to be displayed

Returns

<String> See Arguments

Examples

`SENSE:EMC:LIST` might return `ALL`, indicating All spots are displayed.

SENSe:EMC:MARGin

Sets or queries the Margin value in EMC View.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MARGin <value>
SENSe:EMC:MARGin?
```

Arguments

<value> = <NR3>

Returns

<NR3>

Examples

SENSE:EMC:MARGIN might return 15, indicating that the margin value is 15 dB.

SENSe:EMC:MEASurement:AMBIent:DELTa:THREshold

Sets or queries Delta threshold value of Ambient detection in EMC View.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MEASurement:AMBIent:DELTa:THREshold <value>
SENSe:EMC:MEASurement:AMBIent:DELTa:THREshold?
```

Arguments

<value> = <NRf>

Returns

<NRf>

Examples

SENSE:EMC:MEASUREMENT:AMBIENT:DELTA:THRESHOLD might return 1.00000, indicating that the Delta threshold value of ambient detection is 1.0 dB.

SENSe:EMC:MEASurement:AMBIent:DETEctor

Sets or queries the detector type of the ambient detector in EMC View.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MEASurement:AMBIent:DETEctor <arg>
SENSe:EMC:MEASurement:AMBIent:DETEctor?
```

Arguments

<arg> =

AVER = Sets Avg (VRMS) as the detector type

PEAK = Sets +Peak as the detector type

MILSTD = Sets Mil +Peak as the detector type

QUAS = Sets CISPR QPk as the detector type

CPE = Sets CISPR +Pk as the detector type

CAV = Sets CISPR Avg as the detector type

AVGL = Sets Avg (of logs) as the detector type

CAVL = Sets CISPR Avg (of logs) as the detector type

DSA = Sets DEF STAN Avg as the detector type

DSP = Sets DEF STAN Peak as the detector type

Returns

<String> See Argument

Examples

SENSE:EMC:MEASUREMENT:AMBIENT:DETECTOR might return DSP, indicating that the ambient detector type is DEF STAN Peak.

SENSe:EMC:MEASurement:AMBIent:DWELl:TIME

Sets or queries the dwell time of ambient detection in EMC View.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MEASurement:AMBIent:DWELl:TIME <value>
SENSe:EMC:MEASurement:AMBIent:DWELl:TIME?
```

Arguments

<value> = <NRf>

Returns

<NRf>

Examples

SENSE:EMC:MEASUREMENT:AMBIENT:DWELL:TIME might return 329.142E-6, indicating that the dwell time of ambient detection is 329.1 µs.

SENSe:EMC:MEASurement:AMBIent:DWELl:TIME:AUTO

Sets or queries to automatically set the dwell time of ambient detection in EMC View.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MEASurement:AMBIent:DWELl:TIME:AUTO { 0 | OFF | 1 | ON}
SENSe:EMC:MEASurement:AMBIent:DWELl:TIME:AUTO?
```

Arguments

`0` or `OFF` disables the Auto check box of dwell time

`1` or `ON` enables the Auto check box of dwell time

Returns

<Boolean> 0,1

Examples

`SENSE:EMC:MEASUREMENT:AMBIENT:DWELL:TIME:AUTO?` might return `1`, indicating that the dwell time of ambient detection is set automatically.

SENSe:EMC:MEASurement:AMBIent:STARt (No Query Form)

Starts the ambient detection with respect to selected detector.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MEASurement:AMBIent:STARt
```

Arguments

None

Returns

None

Examples

`SENSE:EMC:MEASUREMENT:AMBIENT:START SINGle` performs a single ambient scan with respect to the selected scan detector.

SENSe:EMC:MEASurement:INSPect:DETEctor<x>

Sets or queries the detector type for the specified detector in discrete search of EMC View.

<x> = 1 or 2 or 3 specifies detector number

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MEASurement:INSPect:DETEctor<x> <arg>
SENSe:EMC:MEASurement:INSPect:DETEctor<x>?
```

Arguments

<arg> =

AVER = Sets Avg (VRMS) as the detector type

PEAK = Sets +Peak as the detector type

MILSTD = Sets Mil +Peak as the detector type

QUAS = Sets CISPR QPk as the detector type

CPE = Sets CISPR +Pk as the detector type

CAV = Sets CISPR Avg as the detector type

AVGL = Sets Avg (of logs) as the detector type

CAVL = Sets CISPR Avg (of logs) as the detector type

DSA = Sets DEF STAN Avg as the detector type

DSP = Sets DEF STAN Peak as the detector type

Returns

<String> See Argument

Examples

SENSe:EMC:MEASurement:INSPect:DETEctor1? might return QUAS, indicating that the detector type for detector 1 in discrete search is CISPR QPk.

SENSe:EMC:MEASurement:INSPect:DETEctor<x>:COMParision:LEVEl

Sets or queries the comparison level value for the specified detector in discrete search of EMC View.

<x> = 1 or 2 or 3 (specifies detector number)

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MEASurement:INSPect:DETEctor<x>:COMParision:LEVEl <value>
SENSe:EMC:MEASurement:INSPect:DETEctor<x>:COMParision:LEVEl?
```

Arguments

<value> = <NRf>

Returns

<NRf>

Examples

SENSe:EMC:MEASurement:INSPect:DETEctor1:COMParision:LEVEl might return 39.9897, indicating that the comparison level value of detector 1 is 39.98 dBuV.

SENSe:EMC:MEASurement:INSPect:DETEctor<x>:ENABle

Sets or queries the Enabling or disabling of specified detector in discrete search of EMC View.

<x> = 1 or 2 or 3 (specifies detector number)

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MEASurement:INSPect:DETEctor<x>:ENABle { 0 | OFF | 1 | ON}
SENSe:EMC:MEASurement:INSPect:DETEctor<x>:ENABle?
```

Arguments

0 or OFF disables the specified detector

1 or ON enables the specified detector

Returns

<Boolean> 0,1

Examples

SENSe:EMC:MEASurement:INSPect:DETEctor1:ENABle might return 1, indicating detector 1 is enabled.

SENSe:EMC:MEASurement:INSPect:DWELl:TIME

Sets or queries the dwell time in discrete search of EMC View.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MEASurement:INSPect:DWELl:TIME <value>
SENSe:EMC:MEASurement:INSPect:DWELl:TIME?
```

Arguments

<value> = <NRf>

Returns

<NRf>

Examples

SENSE:EMC:MEASUREMENT:INSPECT:DWELL:TIME might return 2.500, indicating that the dwell time is 2.5 s.

SENSe:EMC:MEASurement:INSPect:DWELl:TIME:AUTO

Sets or queries to automatically set the dwell time in discrete search of EMC View.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MEASurement:INSPect:DWELl:TIME:AUTO { 0 | OFF | 1 | ON}
SENSe:EMC:MEASurement:INSPect:DWELl:TIME:AUTO?
```

Arguments

0 or OFF disables the auto check box

1 or ON enables the auto check box

Returns

<Boolean> 0,1

Examples

SENSE:EMC:MEASUREMENT:INSPECT:DWELL:TIME:AUTO might return 0, indicating auto check box is disabled.

SENSe:EMC:MEASurement:INSPect:MODE

Sets or queries the inspect search mode in EMC View.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MEASurement:INSPect:MODE { DISC | HARM }
SENSe:EMC:MEASurement:INSPect:MODE?
```

Arguments

DISC = Sets inspect search to Discrete mode

HARM = Sets inspect search to Harmonic mode

Returns

<String> DISC or HARM

Examples

SENSE:EMC:MEASUREMENT:INSPECT:MODE might return HARM, indicating inspect search is in harmonic search mode.

SENSe:EMC:MEASurement:INSPect:MODE:DISCrete:FREQuency<x>:ENABle

Sets or queries the enabling or disabling of the specified row number in discrete search of EMC View.

<x> = <NR1> specifies row number

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MEASurement:INSPect:MODE:DISCrete:FREQuency<x>:ENABle { 0 | OFF | 1 | ON}
SENSe:EMC:MEASurement:INSPect:MODE:DISCrete:FREQuency<x>:ENABle?
```

Arguments

`0` or `OFF` disables the specified row number

`1` or `ON` enables the specified row number

Returns

<Boolean> 0,1

Examples

`SENSe:EMC:MEASurement:INSPect:MODE:DISCrete:FREQuency3:ENABle?` might return `1`, indicating that the row 3 is enabled.

SENSe:EMC:MEASurement:INSPect:MODE:DISCrete:FREQuency<x>:FREQuency

Sets or queries the frequency value of the specified row number in discrete search of EMC View.

<x> = <NR1> specifies row number

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MEASurement:INSPect:MODE:DISCrete:FREQuency<x>:FREQuency <value>
SENSe:EMC:MEASurement:INSPect:MODE:DISCrete:FREQuency<x>:FREQuency?
```

Arguments

`<value>` = <NRf> frequency value

Returns

<NRf>

Examples

SENSe:EMC:MEASurement:INSPect:MODE:DISCrete:FREQuency2:FREQuency? might return 25.00000E+6, indicating that the frequency value of row 2 is 25.00 MHz.

SENSe:EMC:MEASurement:INSPect:MODE:DISCrete:FREQuency<x>:RBW

Sets or queries the RBW value of the specified row number in discrete search of EMC View.

<x> = <NR1> Specifies row number

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MEASurement:INSPect:MODE:DISCrete:FREQuency<x>:RBW <value>
SENSe:EMC:MEASurement:INSPect:MODE:DISCrete:FREQuency<x>:RBW?
```

Arguments

<value> = <NRf> RBW value

Returns

<NRf>

Examples

SENSe:EMC:MEASurement:INSPect:MODE:DISCrete:FREQuency1:RBW? might return 9.00000E+3, indicating that the RBW value of row 1 is 9.00 kHz.

SENSe:EMC:MEASurement:INSPect:MODE:DISCrete:FREQuency<x>:RBW:AUTO

Sets or queries the enabling or disabling of Auto RBW check box of the specified row number in discrete search of EMC View.

<x> = <NR1> specifies row number

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MEASurement:INSPect:MODE:DISCrete:FREQuency<x>:RBW:AUTO { 0 | OFF | 1 | ON}
SENSe:EMC:MEASurement:INSPect:MODE:DISCrete:FREQuency<x>:RBW:AUTO?
```

Arguments

0 or OFF disables the Auto check box

1 or ON enables the Auto check box

Returns

<Boolean> 0,1

Examples

SENSe:EMC:MEASurement:INSPect:MODE:DISCrete:FREQuency4:RBW:AUTO? might return 1, indicating that the Auto check box for row 4 is enabled.

SENSe:EMC:MEASurement:INSPect:MODE:DISCrete:LOAD:EMISsion:RESults (No Query Form)

Loads the frequency spots from Emission results into discrete search table. Only the top 20 spots of emission results will be loaded.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

SENSe:EMC:MEASurement:INSPect:MODE:DISCrete:LOAD:EMISsion:RESults

Arguments

None

Returns

None

Examples

SENSE:EMC:MEASUREMENT:INSPECT:MODE:DISCRETE:LOAD:EMISSION:RESULTS loads the emission results.

SENSe:EMC:MEASurement:INSPect:MODE:HARMonic<x>:ENABle

Sets or queries to enable or disable the specified harmonic in EMC View.

<x> = 1 or 2 or 3 specifies the harmonic

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

SENSe:EMC:MEASurement:INSPect:MODE:HARMonic<x>:ENABle { 0 | OFF | 1 | ON}
SENSe:EMC:MEASurement:INSPect:MODE:HARMonic<x>:ENABle?

Arguments

0 or OFF disables the specified harmonic

1 or ON enables the specified harmonic

Returns

<Boolean> 0,1

Examples

SENSe:EMC:MEASurement:INSPect:MODE:HARMonic1:ENABle? might return 1, indicating that the harmonic 1 in enabled.

SENSe:EMC:MEASurement:INSPect:MODE:HARMonic<x>:FACTor<y>:ENABle

Sets or queries the enabling or disabling harmonic factor of a specified harmonic in EMC View.

<x> = 1 or 2 or 3 specifies the harmonic

<y> = <NR1> (1 to 10) Specifies the harmonic factor

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MEASurement:INSPect:MODE:HARMonic<x>:FACTor<y>:ENABle { 0 | OFF | 1 | ON}
SENSe:EMC:MEASurement:INSPect:MODE:HARMonic<x>:FACTor<y>:ENABle?
```

Arguments

0 or OFF disables the harmonic factor

1 or ON enables the harmonic factor

Returns

<Boolean> 0,1

Examples

SENSe:EMC:MEASurement:INSPect:MODE:HARMonic1:FACTor2:ENABle might return 1, indicating that harmonic factor 2 of harmonic 1 is enabled.

SENSe:EMC:MEASurement:INSPect:MODE:HARMonic<x>:FACTor<y>:FREQuency

Sets or queries the frequency value of the specified harmonic and factor in harmonic search of EMC View.

p<x> = 1 or 2 or 3 specifies the harmonic

<y> = <NR1> (1 to 10) specifies the harmonic factor

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MEASurement:INSPect:MODE:HARMonic<x>:FACTor<y>:FREQuency <value>
SENSe:EMC:MEASurement:INSPect:MODE:HARMonic<x>:FACTor<y>:FREQuency?
```

Arguments

`<value>` = <NRf> Frequency value

Returns

<NRf>

Examples

`SENSe:EMC:MEASurement:INSPect:MODE:HARMonic1:FACTor2:FREQuency?` might return `25.00000E+6`, indicating that the frequency value of harmonic 1 factor 2 is 25.00 MHz.

SENSe:EMC:MEASurement:INSPect:MODE:HARMonic<x>:FACTor<y>:HARMonic:FACTor

Sets or queries the harmonic factor value for the specified harmonic in EMC View.

<x> = 1 or 2 or 3 specifies the harmonic

<y> = <NR1> (1 to 10) Specifies the harmonic factor

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MEASurement:INSPect:MODE:HARMonic<x>:FACTor<y>:HARMonic:FACTor <value>
SENSe:EMC:MEASurement:INSPect:MODE:HARMonic<x>:FACTor<y>:HARMonic:FACTor?
```

Arguments

`<value>` =

Either

`OEIG` = 1/8

`QUAR` = 1/4

`HALF` = 1/2

`ONE` = 1

`TWO` = 2

`THREE` = 3

`FOUR` = 4

`FIVE` = 5

`SIX` = 6

`SEVEN` = 7

EIGHT = 8

NINE = 9

TEN = 10

Returns

<String> See Arguments

Examples

SENSe:EMC:MEASurement:INSPect:MODE:HARMonic1:FACTor2:HARMonic:FACTor? might return "HALF", indicating the 2nd harmonic factor value for harmonic 1 is ½.

SENSe:EMC:MEASurement:INSPect:MODE:HARMonic<x>:FACTor<y>:RBW

Sets or queries the RBW value of the specified harmonic and factor in harmonic search of EMC View.

<x> = 1 or 2 or 3 specifies the harmonic

<y> = <NR1> (1 to 10) specifies the harmonic factor

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MEASurement:INSPect:MODE:HARMonic<x>:FACTor<y>:RBW <value>
SENSe:EMC:MEASurement:INSPect:MODE:HARMonic<x>:FACTor<y>:RBW?
```

Arguments

<value> = <NRf> RBW value

Returns

<NRf>

Examples

SENSe:EMC:MEASurement:INSPect:MODE:HARMonic1:FACTor2:RBW? might return 9.00000E+3, indicating that the RBW value of harmonic 1 factor 2 is 9.00 kHz.

SENSe:EMC:MEASurement:INSPect:MODE:HARMonic<x>:FACTor<y>:RBW:AUTO

Sets or queries the enabling and disabling of Auto RBW check box with respect to the specified harmonic and factor in harmonic search of EMC View.

<x> = 1 or 2 or 3 specifies the harmonic

<y> = <NR1> (1 to 10) specifies the harmonic factor

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MEASurement:INSPect:MODE:HARMonic<x>:FACTor<y>:RBW:AUTO { 0 | OFF | 1 | ON}
SENSe:EMC:MEASurement:INSPect:MODE:HARMonic<x>:FACTor<y>:RBW:AUTO?
```

Arguments

`0` or `OFF` disables the Auto checkbox

`1` or `ON` enables the Auto checkbox

Returns

<Boolean> 0,1

Examples

`SENSe:EMC:MEASurement:INSPect:MODE:HARMonic2:FACTor1:RBW:AUTO?` might return `1`, indicating that the Auto check box is enabled for harmonic 2 factor 1.

SENSe:EMC:MEASurement:INSPect:MODE:HARMonic<x>:FREquency

Sets or queries the frequency of the Specified Harmonic in EMC View.

<x> = 1 or 2 or 3 Specifies the harmonic

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MEASurement:INSPect:MODE:HARMonic<x>:FREquency <value>
SENSe:EMC:MEASurement:INSPect:MODE:HARMonic<x>:FREquency?
```

Arguments

`<value>` = <NRf>

Returns

<NRf>

Examples

`SENSe:EMC:MEASurement:INSPect:MODE:HARMonic1:FREquency` returns `20000`, indicating that the frequency of harmonic 1 is 20 kHz.

SENSe:EMC:MEASurement:INSPect:STARt (No Query Form)

Sets the type of run in inspect section of EMC display.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

```
Syntax
SENSe:EMC:MEASurement:INSPect:STARt {SINGle | CONTinous | STOP}
```

Arguments

`SINGle` = Performs Single run

`CONTinuous` = Performs a Continuous run

`STOP` = Stops the acquisition

Returns

None

Examples

`SENSE:EMC:MEASUREMENT:INSPECT:START SINGle` sets the inspect mode to perform a single run.

SENSe:EMC:MEASurement:SCAN:DETEctor<x>

Sets or queries the detector type of the specified scan detector in EMC View.

<x> = 1 or 2

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MEASurement:SCAN:DETEctor<x> <arg>
SENSe:EMC:MEASurement:SCAN:DETEctor<x>?
```

Arguments

`<arg>` =

`AVER` = Sets Avg (VRMS) as the detector type

`PEAK` = Sets +Peak as the detector type

`MILSTD` = Sets Mil +Peak as the detector type

`QUAS` = Sets CISPR QPk as the detector type

`CPE` = Sets CISPR +Pk as the detector type

`CAV` = Sets CISPR Avg as the detector type

`AVGL` = Sets Avg (of logs) as the detector type

`CAVL` = Sets CISPR Avg (of logs) as the detector type

`DSA` = Sets DEF STAN Avg as the detector type

`DSP` = Sets DEF STAN Peak as the detector type

Returns

<String> See Arguments

Examples

SENSe:EMC:MEASurement:SCAN:DETEctor1? might return QUAS, indicating that the detector type for scan detector 1 is CISPR QPk.

SENSe:EMC:MEASurement:SCAN:DETEctor<x>:ENABle

Sets or queries the enabling or disabling of the specified scan detector in EMC View.

<x> = 1 or 2

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MEASurement:SCAN:DETEctor<x>:ENABle { 0 | OFF | 1 | ON}
SENSe:EMC:MEASurement:SCAN:DETEctor<x>:ENABle?
```

Arguments

0 or OFF disables the scan detector

1 or ON enables the scan detector

Returns

<Boolean> 0,1

Examples

SENSe:EMC:MEASurement:SCAN:DETEctor1:ENABle? might return 1, indicating that scan detector 1 is enabled.

SENSe:EMC:MEASurement:SCAN:DETEctor<x>:LIMit

Sets or queries the limit of specified scan detector in EMC View.

<x> = 1 or 2

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MEASurement:SCAN:DETEctor<x>:LIMit <arg>
SENSe:EMC:MEASurement:SCAN:DETEctor<x>:LIMit?
```

Arguments

<arg> = Either ONE , TWO or THREE

ONE = Sets Limit to Limit 1

TWO = Sets Limit to Limit 2

THREE = Sets Limit to Limit 3

Returns

<String>See Argument

Examples

SENSe:EMC:MEASurement:SCAN:DETEctor1:LIMit? might return TWO, indicating that the limit selected for scan detector 1 is Limit 2.

SENSe:EMC:MEASurement:SCAN:DWELI:TIME

Sets or queries the dwell time of scan detection in EMC View.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MEASurement:SCAN:DWELl:TIME <value>
SENSe:EMC:MEASurement:SCAN:DWELl:TIME?
```

Arguments

<value> = <NRf>

Returns

<NRf>

Examples

SENSE:EMC:MEASUREMENT:SCAN:DWELL:TIME? might return 640.315E-3, indicating the dwell time of scan detection is 640.3 ms.

SENSe:EMC:MEASurement:SCAN:DWELI:TIME:AUTO

Sets or queries to automatically set the dwell time of scan detection in EMC View.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MEASurement:SCAN:DWELl:TIME:AUTO { 0 | OFF | 1 |ON}
SENSe:EMC:MEASurement:SCAN:DWELl:TIME:AUTO?
```

Arguments

0 or OFF disables the Auto check box of dwell time

1 or ON enables the Auto check box of dwell time

Returns

<Boolean> 0,1

Examples

SENSE:EMC:MEASUREMENT:SCAN:DWELL:TIME:AUTO? might return 1, indicating that the dwell time of scan detection is set automatically.

SENSe:EMC:MEASurement:SCAN:STARt (No Query Form)

Starts the scan on the signal with respect to selected scan detector and displays the measurement values.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MEASurement:SCAN:STARt {SINGle | CONTinuous | STOP}
```

Arguments

SINGle = Performs a Single scan

CONTinuous = Performs a Continuous scan

STOP = Stops the acquisition

Returns

None

Examples

SENSE:EMC:MEASUREMENT:SCAN:START SINGle performs a Single scan with respect to the selected scan detector.

SENSe:EMC:MEASurement:SPOT:ALL

Sets or queries to enable or disable All check box to include all spots for spot analysis.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MEASurement:SPOT:ALL { 0 | OFF | 1 | ON}
SENSe:EMC:MEASurement:SPOT:ALL?
```

Arguments

0 or OFF disables the All check box

1 or ON enables the All check box

Returns

<Boolean> 0,1

Examples

SENSE:EMC:MEASUREMENT:SPOT:ALL? might return 1, indicating that all spots are included for spot analysis.

SENSe:EMC:MEASurement:SPOT:COUNt

Sets or queries the Maximum number of spots required for spot Analysis in EMC View. Applies only to Pre-scan + Auto spot measurement type.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MEASurement:SPOT:COUNt <value>
SENSe:EMC:MEASurement:SPOT:COUNt?
```

Arguments

<value> = <NRf>

Returns

<NRf>

Examples

SENSE:EMC:MEASUREMENT:SPOT:COUNT? might return 20, indicating 20 spots to be selected for spot analysis.

SENSe:EMC:MEASurement:SPOT:DETEctor<x>

Sets or queries the detector type of the specified spot detector in EMC View.

<x> = 1 or 2 or 3

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MEASurement:SPOT:DETEctor<x> <arg>
SENSe:EMC:MEASurement:SPOT:DETEctor<x>?
```

Arguments

<arg> =

AVER = Sets Avg (VRMS) as the detector type

PEAK = Sets +Peak as the detector type

MILSTD = Sets Mil +Peak as the detector type

QUAS = Sets CISPR QPk as the detector type

CPE = Sets CISPR +Pk as the detector type

CAV = Sets CISPR Avg as the detector type

AVGL = Sets Avg (of logs) as the detector type

CAVL = Sets CISPR Avg (of logs) as the detector type

DSA = Sets DEF STAN Avg as the detector type

DSP = Sets DEF STAN Peak as the detector type

Returns

<String> See Argument

Examples

SENSe:EMC:MEASurement:SPOT:DETEctor1? might return QUAS, indicating that the spot detector type for detector 1 is CISPR QPk.

SENSe:EMC:MEASurement:SPOT:DETEctor<x>:ENABle

Sets or queries the enabling and disabling of specified spot detector in EMC View.

<x> = 1 or 2 or 3

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MEASurement:SPOT:DETEctor<x>:ENABle { 0 | OFF | 1 | ON}
SENSe:EMC:MEASurement:SPOT:DETEctor<x>:ENABle?
```

Arguments

0 or OFF disables the spot detector

1 or ON enables the spot detector

Returns

<Boolean> 0,1

Examples

SENSe:EMC:MEASurement:SPOT:DETEctor1:ENABle? might return 1, indicating that spot detector 1 is enabled.

SENSe:EMC:MEASurement:SPOT:DETEctor<x>:LIMit

Sets or queries the limit of specified spot detector in EMC View.

<x> = 1 or 2 or 3

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MEASurement:SPOT:DETEctor<x>:LIMit <arg>
SENSe:EMC:MEASurement:SPOT:DETEctor<x>:LIMit?
```

Arguments

<arg> = Either ONE , TWO or THREE

ONE = Sets Limit to Limit 1

TWO = Sets Limit to Limit 2

THREE = Sets Limit to Limit 3

Returns

<String> See Argument

Examples

SENSe:EMC:MEASurement:SPOT:DETEctor2:LIMit? might return ONE, indicating that the limit selected for spot detector 2 is Limit 1.

SENSe:EMC:MEASurement:SPOT:DWELI:TIME

Sets or queries the dwell time of spot detection in EMC View.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MEASurement:SPOT:DWELl:TIME <value>
SENSe:EMC:MEASurement:SPOT:DWELl:TIME?
```

Arguments

<value> = <NRf>

Returns

<NRf>

Examples

SENSE:EMC:MEASUREMENT:SPOT:DWELL:TIME? might return 1.000000, indicating that the dwell time of spot detection is 1 s.

SENSe:EMC:MEASurement:SPOT:DWELl:TIME:AUTO

Sets or queries to automatically set the dwell time of spot detection in EMC View.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MEASurement:SPOT:DWELl:TIME:AUTO { 0 | OFF | 1 | ON}
SENSe:EMC:MEASurement:SPOT:DWELl:TIME:AUTO?
```

Arguments

`0` or `OFF` disables the Auto check box of dwell time

`1` or `ON` enables the Auto check box of dwell time

Returns

<Boolean> 0,1

Examples

`SENSE:EMC:MEASUREMENT:SPOT:DWELL:TIME:AUTO?` might return `1`, indicating that the dwell time of spot detection is set automatically.

SENSe:EMC:MEASurement:SPOT:STARt (No Query Form)

Starts the spot re-measurement on the selected spots with respect to selected spot detector.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MEASurement:SPOT:STARt
```

Arguments

None

Returns

None

Examples

`SENSE:EMC:MEASUREMENT:SPOT:START` starts spot re-measurement on the selected spots.

SENSe:EMC:MEASurement:TYPE

Sets or queries the measurement type in EMC View.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MEASurement:TYPE {PSMS | PSAS | CS}
SENSe:EMC:MEASurement:TYPE?
```

Arguments

`PSMS` = Sets Pre-scan+ Manual Spot as measurement type.

`PSAS` = Sets Pre-scan+ Auto Spot as measurement type.

`CS` = Sets Complete scan as measurement type.

Returns

<String> PSMS or PSAS or CS

Examples

`SENSE:EMC:MEASUREMENT:TYPE` might return `CS`, indicating that Complete scan is the measurement type selected.

SENSe:EMC:MODE

Sets or queries the frequency ranges (mode) in EMC view.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:MODE {MULT | SING}
SENSe:EMC:MODE?
```

Arguments

`MULT` = Sets multiple as frequency range

`SING` = Sets Single as frequency range

Returns

<String> MULT or SING

Examples

`SENSE:EMC:MODE?` might return `SING`, indicating that the frequency range is Single.

SENSe:EMC:OPTimization

Sets or queries the method of optimizing the gain and input bandwidth in the EMC display.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:OPTimization { AUTO | MINTime | MAXDynrange | MINNoise }
SENSe:EMC:OPTimization?
```

Arguments

`AUTO` = optimizes automatically the gain and input bandwidth.

`MINTime` = optimizes the gain and input bandwidth to minimize sweep time.

`MAXDynrange` = optimizes the gain and input bandwidth to maximize the dynamic range.

`MINNoise` = optimizes the gain and input bandwidth to minimize noise.

Returns

<String> See Argument

Examples

`SENSe:EMC:OPTIMIZATION AUTO` optimizes automatically the gain and input bandwidth.

SENSe:EMC:POINts:COUNt

Sets or queries the trace points value in EMC view.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:POINts:COUNt {P801 | P2401 | P4001 | P8001 | P10401 | P16001 | P32001 | P64001}
SENSe:EMC:POINts:COUNt?
```

Arguments

`P801` = Sets 801 as trace point value.

`P2401` = Sets 2401 as trace point value.

`P4001` = Sets 4001 as trace point value.

`P8001` = Sets 8001 as trace point value.

`P10401` = Sets 10401 as trace point value.

`P16001` = Sets 16001 as trace point value.

`P32001` = Sets 32001 as trace point value.

`P64001` = Sets 64001 as trace point value.

Returns

<String> See Arguments

Examples

`SENSe:EMC:POINTS:COUNT P2401` sets `2401` as trace point value in EMC view.

SENSe:EMC:RANGe<x>:BANDwidth:VIDeo

Sets or queries the (Video Bandwidth) VBW value of selected range in EMC View.

<x> = <NR1> 1 to 20

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:RANGe<x>:BANDwidth:VIDeo <value>
SENSe:EMC:RANGe<x>:BANDwidth:VIDeo?
```

Arguments

<value> = <NR3>

Returns

<NR3>

Examples

`SENSe:EMC:RANGe1:BANDwidth:VIDeo?` might return `1.0000000E+6`, indicating that the VBW value of range 1 is 1 MHz.

SENSe:EMC:RANGe<x>:BANDwidth:VIDeo:STATe

Sets or queries the enabling or disabling the (Video state) VBW check box of specified range in EMC View.

<x> = <NR1> 1 to 20

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:RANGe<x>:BANDwidth:VIDeo:STATe { 0 | OFF | 1 | ON}
SENSe:EMC:RANGe<x>:BANDwidth:VIDeo:STATe?
```

Arguments

`0` or `OFF` disables the VBW check box

`1` or `ON` enables the VBW check box

Returns

<Boolean> 0,1

Examples

`SENSe:EMC:RANGe2:BANDwidth:VIDeo:STATe?` might return `0`, indicating that VBW check box is disabled for range 2.

SENSe:EMC:RANGe<x>:EXCursion

Sets or queries the excursion value of the specified range in EMC View.

<x> = <NR1> 1 to 20

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:RANGe<x>:EXCursion <value>
SENSe:EMC:RANGe<x>:EXCursion?
```

Arguments

<value> = <NR3>

Returns

<NR3>

Examples

`SENSe:EMC:RANGe1:EXCursion?` might return `6.0000`, indicating that the threshold value of range 1 is 6 dB.

SENSe:EMC:RANGe<x>:FILTer[:SHAPe]:BANDwidth

Sets or queries the (Resolution Bandwidth) RBW value of selected range in EMC View.

<x> = <NR1> 1 to 20

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:RANGe<x>:FILTer[:SHAPe]:BANDwidth <value>
SENSe:EMC:RANGe<x>:FILTer[:SHAPe]:BANDwidth?
```

Arguments

<value> = <NR3>

Returns

<NR3>

Examples

`SENSe:EMC:RANGe1:FREQuency:STOP?` might return `9.00`, indicating that the RBW value of range 1 is 9 kHz.

SENSe:EMC:RANGe<x>:FILTer[:SHAPe]:BANDwidth:AUTO

Sets or queries the Enabling or disabling of RBW Auto checkbox of selected range to set RBW value automatically in EMC View.

<x> = <NR1> 1 to 20

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:RANGe<x>:FILTer[:SHAPe]:BANDwidth:AUTO { 0 | OFF |1 | ON}
SENSe:EMC:RANGe<x>:FILTer[:SHAPe]:BANDwidth:AUTO?
```

Arguments

`0` or `OFF` disables the Auto check box

`1` or `ON` enables the Auto check box

Returns

<Boolean> 0,1

Examples

`SENSe:EMC:RANGe0:FREQuency:STOP?` might return `0`, indicating that Auto check box is disabled.

SENSe:EMC:RANGe<x>:FREQuency:STARt

Sets or queries the Start frequency of the specified range in EMC View.

<x> = <NR1> 1 to 20

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:RANGe<x>:FREQuency:STARt <value>
SENSe:EMC:RANGe<x>:FREQuency:STARt?
```

Arguments

<value> = <NR3>

Returns

<NR3>

Examples

`SENSe:EMC:RANGe1:FREQuency:STARt?` might return `150000`, indicating that the start frequency of range 1 is 150 kHz.

SENSe:EMC:RANGe<x>:FREQuency:STOP

Sets or queries the stop frequency of selected range in EMC View.

<x> = <NR1> 1 to 20

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:RANGe<x>:FREQuency:STOP <value>
SENSe:EMC:RANGe<x>:FREQuency:STOP?
```

Arguments

<value> = <NR3>

Returns

<NR3>

Examples

`SENSe:EMC:RANGe1:FREQuency:STOP?` might return `500000`, indicating that the stop frequency of range 1 is 500 kHz.

SENSe:EMC:RANGe<x>:LIMit<y>:EQUAl

Sets or queries whether limit start and stop value of selected range and limit are the same in EMC View.

<x> = <NR1> 1 to 20

<y> = <NR1> Either Limit 1 or 2 or 3

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:RANGe<x>:LIMit<y>:EQUAl { 0 | OFF | 1 | ON}
SENSe:EMC:RANGe<x>:LIMit<y>:EQUAl?
```

Arguments

`0` or `OFF` disables the Limit same check box

`1` or `ON` enables the Limit same check box

Returns

<Boolean> 0,1

Examples

`SENSe:EMC:RANGe1:LIMit1:EQUAl?` might return `1`, indicating the Limit 1 start and stop value for range 2 are the same.

SENSe:EMC:RANGe<x>:LIMit<y>:STARt

Sets or queries the Limit start value of the specified range and specified limit in EMC View.

<x> = <NR1> 1 to 20

<y> = <NR1> Either Limit 1 or 2 or 3

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:RANGe<x>:LIMit<y>:STARt <value>
SENSe:EMC:RANGe<x>:LIMit<y>:STARt?
```

Arguments

<value> = <NR3>

Returns

<NR3>

Examples

`SENSe:EMC:RANGe1:LIMit1:STARt?` might return `79.0000`, indicating that the Limit 1 start value of range 1 is 79 dBuV.

SENSe:EMC:RANGe<x>:LIMit<y>:STOP

Sets or queries the Limit stop value of the specified range and specified limit in EMC View.

<x> = <NR1> 1 to 20

<y> = <NR1> Either Limit 1 or 2 or 3

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:RANGe<x>:LIMit<y>:STOP <value>
SENSe:EMC:RANGe<x>:LIMit<y>:STOP?
```

Arguments

<value> = <NR3>

Returns

<NR3>

Examples

`SENSe:EMC:RANGe1:LIMit1:STOP?` might return `79.0000`, indicating that the Limit 1 stop value of range 1 is 79 dBuV.

SENSe:EMC:RANGe<x>:STATe

Sets or queries Enabling and disabling the selected range in EMC View.

<x> = <NR1> 1 to 20

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:RANGe<x>:STATe { 0 | OFF | 1 | ON}
SENSe:EMC:RANGe<x>:STATe?
```

Arguments

`0` or `OFF` disables the range

`1` or `ON` enables the range

Returns

<Boolean> 0,1

Examples

`SENSe:EMC:RANGe2:STATe?` might return `0`, indicating that range 2 is disabled.

SENSe:EMC:RANGe<x>:THReshold

Sets or queries the threshold value of the specified range in EMC View.

<x> = <NR1> 1 to 20

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:RANGe<x>:THReshold <value>
SENSe:EMC:RANGe<x>:THReshold?
```

Arguments

<value> = <NR3>

Returns

<NR3>

Examples

`SENSe:EMC:RANGe1:THReshold?` might return `46.0000`, indicating that the threshold value of range 1 is 46 dBuV.

SENSe:EMC:REGion

Sets or queries the region of standard in EMC view.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:REGion {EURO | US}
SENSe:EMC:REGion?
```

Arguments

`EURO` = Sets Europe as the region

`US` = Sets US as the region

Returns

<String> See Argument

Examples

`SENSe:EMC:REGION?` might return `US`, indicating that the region selected is US.

SENSe:EMC:REPOrt:INCLude:CONTrol:SETTings

Sets or queries to include or exclude control settings in report by enabling or disabling include control settings check box in report setup of EMC display.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:REPOrt:INCLude:CONTrol:SETTings { 0 | OFF | 1 | ON}
SENSe:EMC:REPOrt:INCLude:CONTrol:SETTings?
```

Arguments

`0` or `OFF` exclude the control settings in report

`1` or `ON` include the control settings in report

Returns

<Boolean> 0,1

Examples

`SENSe:EMC:REPORT:INCLUDE:CONTROL:SETTINGS?` might return `1`, indicating that control settings is included in report.

SENSe:EMC:REPOrt:INCLude:EMISsion:RESUlts

Sets or queries to include or exclude emission results in report by enabling or disabling include emission results check box in report setup of EMC display.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:REPOrt:INCLude:EMISsion:RESUlts { 0 | OFF | 1 | ON}
SENSe:EMC:REPOrt:INCLude:EMISsion:RESUlts?
```

Arguments

`0` or `OFF` exclude emission results in report

`1` or `ON` include emission results in report

Returns

<Boolean> 0,1

Examples

`SENSe:EMC:REPORT:INCLUDE:EMISSION:RESULTS?` might return `1`, indicating that emission results is included in report.

SENSe:EMC:REPOrt:INCLude:ENVIronment:INFOrmation

Sets or queries to include or exclude environment information in report by enabling or disabling environment information check box in report setup of EMC display.

<x> = <NR1> 1 to 20

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:REPOrt:INCLude:ENVIronment:INFOrmation { 0 | OFF | 1 | ON}
SENSe:EMC:REPOrt:INCLude:ENVIronment:INFOrmation?
```

Arguments

`0` or `OFF` exclude environment information in report

1 or ON include environment information in report

Returns

<Boolean> 0,1

Examples

SENSe:EMC:REPORT:INCLUDE:ENVIRONMENT:INFORMATION? might return 1, indicating that environment information is included in report.

SENSe:EMC:REPOrt:INCLude:ENVIronment:INFOrmation:ADD (No Query Form)

Adds one Environment information content by adding one title and its corresponding value in the report setup of EMC display.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

SENSe:EMC:REPOrt:INCLude:ENVIronment:INFOrmation:ADD "Title","Value"

Arguments

Title = <String>

Value = <String>

Returns

None

Examples

SENSe:EMC:REPORT:INCLUDE:ENVIRONMENT:INFORMATION:ADD "Temperature","20" adds the environment information temperature and its value as 20 in the report.

SENSe:EMC:REPOrt:INCLude:ENVIronment:INFOrmation:DELete:ALL (No Query Form)

Deletes all the environment information in the report setup of EMC display.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

SENSe:EMC:REPOrt:INCLude:ENVIronment:INFOrmation:DELete:ALL

Arguments

None

Returns

None

Examples

`SENSe:EMC:REPORT:INCLUDE:ENVIRONMENT:INFORMATION:DELETE:ALL` deletes all the environment information in the report.

SENSe:EMC:REPOrt:INCLude:EUT:INFOrmation

Sets or queries to include or exclude EUT (Equipment under test) information in report by enabling or disabling EUT information check box in report setup of EMC display.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:REPOrt:INCLude:EUT:INFOrmation { 0 | OFF | 1 | ON}
SENSe:EMC:REPOrt:INCLude:EUT:INFOrmation?
```

Arguments

`0` or `OFF` exclude EUT information in report

`1` or `ON` include EUT information in report

Returns

<Boolean> 0,1

Examples

`SENSe:EMC:REPORT:INCLUDE:EUT:INFORMATION?` might return `0`, indicating that EUT information is excluded in report.

SENSe:EMC:REPOrt:INCLude:EUT:INFOrmation:ADD (No Query Form)

Adds one EUT information content by adding one tittle and its corresponding value in the report setup of EMC display.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:REPOrt:INCLude:EUT:INFOrmation:ADD "Title","Value"
```

Arguments

Title = <String>

Value = <String>

Returns

None

Examples

`SENSe:EMC:REPORT:INCLUDE:EUT:INFORMATION:ADD "Serial Number","ABC156"` adds the EUT information Serial Number and its value as ABC156 in the report.

SENSe:EMC:REPOrt:INCLude:EUT:INFOrmation:DELete:ALL (No Query Form)

Deletes all the EUT information in the report setup of EMC display.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

`SENSe:EMC:REPOrt:INCLude:EUT:INFOrmation:DELete:ALL`

Arguments

None

Returns

None

Examples

`SENSe:EMC:REPORT:INCLUDE:EUT:INFORMATION:DELETE:ALL` deletes all the EUT information in the report.

SENSe:EMC:REPOrt:INCLude:GRAPh

Sets or queries to include or exclude graph of measurement results in report by enabling or disabling include graph check box in report setup of EMC display.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:REPOrt:INCLude:GRAPh { 0 | OFF | 1 | ON}
SENSe:EMC:REPOrt:INCLude:GRAPh?
```

Arguments

0 or OFF exclude the graph in report

1 or ON include the graph in report

Returns

<Boolean> 0,1

Examples

SENSe:EMC:REPORT:INCLUDE:GRAPH? might return 0, indicating that graph is excluded in report.

SENSe:EMC:REPOrt:INCLude:INSPect:FREQuency:RESUlts

Sets or queries to include or exclude inspect frequency results in report by enabling or disabling include inspect req results check box in report setup of EMC display.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:REPOrt:INCLude:INSPect:FREQuency:RESUlts { 0 | OFF | 1 | ON}
SENSe:EMC:REPOrt:INCLude:INSPect:FREQuency:RESUlts?
```

Arguments

0 or OFF exclude inspect frequency results in report

1 or ON include inspect frequency results in report

Returns

<Boolean> 0,1

Examples

SENSe:EMC:REPORT:INCLUDE:INSPECT:FREQUENCY:RESULTS? might return 0, indicating that inspect frequency results is excluded in report.

SENSe:EMC:REPOrt:INCLude:USER:INFOrmation

Sets or queries to include or exclude user information in report by enabling or disabling user information check box in report setup of EMC display.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:REPOrt:INCLude:USER:INFOrmation { 0 | OFF | 1 | ON}
SENSe:EMC:REPOrt:INCLude:USER:INFOrmation?
```

Arguments

0 or OFF exclude user information in report

1 or ON include user information in report

Returns

<Boolean> 0,1

Examples

SENSe:EMC:REPORT:INCLUDE:USER:INFORMATION? might return 1, indicating that user information is included in report.

SENSe:EMC:REPOrt:INCLude:USER:INFOrmation:ADD (No Query Form)

Adds one user information content by adding one title and its corresponding value in the report setup of EMC display.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

SENSe:EMC:REPOrt:INCLude:USER:INFOrmation:ADD "Title","Value"

Arguments

Title = <String>

Value = <String>

Returns

None

Examples

SENSe:EMC:REPORT:INCLUDE:USER:INFORMATION:ADD "Engineer","xyz" adds the user information engineer and its value as xyz in the report.

SENSe:EMC:REPOrt:INCLude:USER:INFOrmation:DELete:ALL (No Query Form)

Deletes all the user information in the report setup of EMC display.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

SENSe:EMC:REPOrt:INCLude:USER:INFOrmation:DELete:ALL

Arguments

None

Returns

None

Examples

SENSe:EMC:REPORT:INCLUDE:USER:INFORMATION:DELETE:ALL deletes all the user information in the report.

SENSe:EMC:REPOrt:MEASurement:HEADing

Sets or queries the measurement heading in the report setup of EMC display.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:REPOrt:MEASurement:HEADing <value>
SENSe:EMC:REPOrt:MEASurement:HEADing?
```

Arguments

<value> = <String>

Returns

<String>

Examples

SENSe:EMC:REPORT:MEASUREMENT:HEADING? might return “spot”, indicating that the measurement heading of the report is spot.

SENSe:EMC:STANdard

Sets or queries the standard in EMC view.

Conditions

Measurement view: EMC - EMI Display

Group

Sense commands

Syntax

```
SENSe:EMC:STANdard <arg>
SENSe:EMC:STANdard?
```

Arguments

When the region is US

<arg> = {FCC Part 15 | FCC Part 18 | MIL-STD 461G}

When the region is Europe

<arg> = {EN55011 | EN55013| EN55014 | EN55015 | EN55025 | EN55032 | EN60601 | DEF STAN | EN61000}

Returns

<String> See Argument

Examples

SENSe:EMC:STANDARD FCC Part 15 sets FCC Part 15 as the standard.

[SENSe]:FDVTime[:MEASview<y>]:SELEcted:SOURce

Sets or queries the source of Frequency deviation versus Time for a selected measurement view number.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Frequency deviation versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Frequency deviation versus Time).

The source cannot be changed when acquisition or analysis is in progress. Stop the acquisition and change the Source.

Group

Sense commands

Syntax

```
[SENSe]:FDVTime[:MEASview<y>]:SELEcted:SOURce {"Source 1" | "Source 2" | "Source 3" | "Source 4"}
[SENSe]:FDVTime[:MEASview<y>]:SELEcted:SOURce?
```

Arguments

The parameter <y>=1 to 8 represents the measurement index based on the number of sources of connected instrument, where the MEASview keyword is optional by default and y is 1.

Examples

SENSe:FDVTime:MEASview<y>:SELECTED:SOURce? queries the Source for Frequency deviation versus Time measurement view 1.

[SENSe]:{FM|PM}:BURSt:THReshold

Sets or queries the threshold level above which the input signal is determined to be a burst in the FM/PM measurement. The burst detected first is used for the measurement.

Conditions

Measurement views: FM, PM

Group

Sense commands

Syntax

```
[SENSe]:{FM|PM}:BURSt:THReshold <value>
[SENSe]:{FM|PM}:BURSt:THReshold?
```

Arguments

`<value>::<NRf>` specifies the threshold level. Range: –100.0 to –10.0 dB.

Returns

<Boolean> 0,1

Examples

`SENSE:FM:BURST:THRESHOLD -10` sets the threshold level to –10 dB in the FM measurement.

[SENSe]:{FM|PM}:FREQuency:OFFSet

Sets or queries the carrier frequency offset in the FM/PM measurement.

Conditions

Measurement views: FM, PM

Group

Sense commands

Syntax

```
[SENSe]:{FM|PM}:FREQuency:OFFSet <value>
[SENSe]:{FM|PM}:FREQuency:OFFSet?
```

Arguments

`<value>::<NRf>` specifies the frequency offset from the center frequency. Range: -(MeasBW*1.1)/2 to +(MeasBW*1.1)/2 where MeasBW is set by the command `[SENSe]:{AM|FM|PM}:{BANDwidth|BWIDth}:MEASurement`.

Examples

`SENSE:FM:FREQUENCY:OFFSET 10MHZ` sets the carrier frequency offset to 10 MHz.

[SENSe]:{FM|PM}:FREQuency:OFFSet:MARKer (No Query Form)

Sets the frequency offset from the selected marker location in the FM measurement.

Sets the frequency offset from the selected delta marker location in the PM measurement.

Conditions

Measurement views: FM, PM

Group

Sense commands

Syntax

```
[SENSe]:{FM|PM}:FREQuency:OFFSet:MARKer
```

Arguments

None

Examples

SENSE:FM:FREQUENCY:OFFSET:MARKER sets frequency offset from the selected Marker position.

[SENSe]:{FM|PM}:FREQuency:SEARch:AUTO

Determines whether to detect the carrier frequency automatically or manually in the FM/PM measurements.

Conditions

Measurement views: FM, PM

Group

Sense commands

Syntax

```
[SENSe]:{FM|PM}:FREQuency:SEARch:AUTO { OFF | ON | 0 | 1 }
[SENSe]:{FM|PM}:FREQuency:SEARch:AUTO?
```

Arguments

OFF or 0 specifies that the carrier frequency is set manually. Use the command [SENSe]:{FM|PM}:FREQuency:OFFSet to set the carrier offset frequency.

ON or 1 specifies that the carrier frequency is detected automatically.

Examples

SENSE:FM:FREQUENCY:SEARCH:AUTO ON specifies that the carrier frequency is detected automatically.

[SENSe]:{FSETtling|PSETtling}:{BANDwidth|BWIDTH}

Sets or queries the measurement bandwidth for the Settling Time measurements.

Conditions

Measurement views: Frequency and Phase Settling Time

Group

Sense commands

Syntax

```
[SENSe]:{FSETtling|PSETtling}:{BANDwidth|BWIDTH} <value>
[SENSe]:{FSETtling|PSETtling}:{BANDwidth|BWIDTH}?
```

Related Commands

FETCh:{AM|FM|PM}? (on page 1062)

READ:AM:RESult? (on page 1557)

Arguments

<value>::<NRf> specifies the bandwidth for the specified settling time measurement.

Examples

SENSE:FSETTLING:BANDwidth 1e6 sets the Frequency Settling time measurement bandwidth to 1 MHz.

[SENSe]:{FSETtling|PSETtling}:{BANDwidth|BWIDTH}:ACTual? (Query Only)

Queries the actual measurement bandwidth in the Settling Time measurements.

Conditions

Measurement views: Frequency and Phase Settling Time

Group

Sense commands

Syntax

```
[SENSe]:{FSETtling|PSETtling}:{BANDwidth|BWIDTH}:ACTual?
```

Arguments

None

Returns

<NRf> The actual measurement bandwidth in Hz.

Examples

SENSE:FSETTLING:BANDWIDTH:ACTUAL? might return 40.000E+6, indicating that the actual RBW is 40.00 MHz.

[SENSe]:{FSETtling|PSETtling}:FREQuency:CENTer

Sets or queries the measurement frequency.

Conditions

Measurement views: Frequency and Phase Settling Time

Group

Sense commands

Syntax

```
[SENSe]:{FSETtling|PSETtling}:FREQuency:CENTer <value>
[SENSe]:{FSETtling|PSETtling}:FREQuency:CENTer?
```

Arguments

<value>::<NRf> specifies the measurement frequency.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples

SENSe:PSETTLING:FREQUENCY:CENTER 7.5GHz sets the Phase Settling Time measurement frequency to 7.5 GHz.

[SENSe]:{FSETtling|PSETtling}:FREQuency:OFFSet

Sets or queries the offset frequency.

Conditions

Measurement views: Frequency and Phase Settling Time

Group

Sense commands

Syntax

```
[SENSe]:{FSETtling|PSETtling}:FREQuency:OFFSet <value>
[SENSe]:{FSETtling|PSETtling}:FREQuency:OFFSet?
```

Arguments

`<value>::<NRf>` specifies the offset frequency.

Range: ± (0.5 × Measurement bandwidth)

Default: 0 Hz.

Examples

`SENSe:PSETTLING:FREQUENCY:OFFSET 20E6` sets the Phase Settling Time offset frequency to 20 MHz.

[SENSe]:{FSETtling|PSETtling}:LENGth

Sets or queries the measurement length.

Conditions

Measurement views: Frequency and Phase Settling Time

Group

Sense commands

Syntax

```
[SENSe]:{FSETtling|PSETtling}:LENGth <value>
[SENSe]:{FSETtling|PSETtling}:LENGth?
```

Arguments

`<value>::<NRf>` specifies the measurement length in seconds.

Range: 10 ns to [(acquisition length) - 400 ns].

If [(analysis start) + (analysis length)] > [(acquisition length) - 400 ns], the actual measurement length is reduced to [(acquisition length) - 200 ns].

Default: 1 ms.

Examples

`SENSe:PSETTLING:LENGTH 25.625us` sets the measurement length to 25.625 µs.

[SENSe]:{FSETtling|PSETtling}:LENGth:ACTual? (Query Only)

Queries the actual measurement length for the Settling Time measurements.

Conditions

Measurement views: Frequency and Phase Settling Time

Group

Sense commands

Syntax

```
[SENSe]:{FSETtling|PSETtling}:LENGth:ACTual?
```

Arguments

None

Returns

<NRf> The actual measurement length in Hz.

Examples

SENSe:FSETTLING:LENGTH:ACTUAL? might return 25.625E-6, indicating that the actual measurement length is 25.625 µs.

[SENSe]:{FSETtling|PSETtling}:MASK:LIMit

Sets or queries the frequency or phase limit for the specified region.

Conditions

Measurement views: Frequency and Phase Settling Time

Group

Sense commands

Syntax

```
[SENSe]:{FSETtling|PSETtling}:MASK:LIMit <x> <value>
[SENSe]:{FSETtling|PSETtling}:MASK:LIMit <x>?
```

Arguments

<x>::= { 1 | 2 | 3} specifies the mask region.

<value>::<NRf> specifies the frequency or phase limit for the region.

For Frequency Settling Time:

- Range: 1 Hz to 1 GHz
- Default: 500 kHz

For Phase Settling Time:

- Range: 0.1 degree to 180 degrees
- Default: 1 degree

Examples

SENSe:FSETTLING:MASK:LIMIT2 7.5E5 sets the limit for region 2 to 750 kHz.

[SENSe]:{FSETtling|PSETtling}:MASK:STATe

Sets or queries mask test state.

Conditions

Measurement views: Frequency and Phase Settling Time

Group

Sense commands

Syntax

```
[SENSe]:{FSETtling|PSETtling}:MASK:STATe { OFF | ON | 0 | 1 }
```

Arguments

`OFF` or `0` enables Mask test.

`ON` or `1` disables Mask test.

Examples

`SENSe:FSETTLING:MASK:STATE ON` enables mask test.

[SENSe]:{FSETtling|PSETtling}:MASK:TIME:REFerence

Sets or queries the mask time reference.

Conditions

Measurement views: Frequency and Phase Settling Time

Group

Sense commands

```
Syntax
[SENSe]:{FSETtling|PSETtling}:MASK:TIME:REFerence { TRIGger | TRANsition }
```

Arguments

`TRIGger` specifies that the trigger point service as the mask time reference.

`TRANsition` specifies that the point where the signal transitions out of the previous tolerance range serves as the mask time reference.

Default: Trigger

Examples

`SENSe:FSETTLING:MASK:TIME:REFerence TRIGGER` specifies that the trigger point serve as the mask time reference.

[SENSe]:{FSETtling|PSETtling}:MASK:TIME:STARt

Sets or queries the mask region start time in seconds.

Conditions

Measurement views: Frequency and Phase Settling Time

Group

Sense commands

Syntax

```
[SENSe]:{FSETtling|PSETtling}:MASK:TIME:STARt <x> <value>
[SENSe]:{FSETtling|PSETtling}:MASK:TIME:STARt <x>?
```

Arguments

`<x>::= { 1 | 2 | 3}` specifies the mask region.

`<value>::<NRf>` specifies the start time of the mask region in seconds.

Examples

`SENSe:FSETTLING:MASK:TIME:START3?` might return `1.0000E-3` as the start time for mask region 3.

[SENSe]:{FSETtling|PSETtling}:MASK:TIME:STOP

Sets or queries the mask region stop time in seconds.

Conditions

Measurement views: Frequency and Phase Settling Time

Group

Sense commands

Syntax

```
[SENSe]:{FSETtling|PSETtling}:MASK:TIME:STOP <value>
[SENSe]:{FSETtling|PSETtling}:MASK:TIME:STOP <x>?
```

Arguments

`<value>::<NRf>` specifies the stop time of the mask in seconds.

Examples

`SENSe:FSETTLING:MASK:TIME:STOP?` might return `6.244120000E-3` as the mask stop time.

[SENSe]:{FSETtling|PSETtling}:MAXTracepoints

Sets or queries the maximum trace points in the Settling Time measurements.

Conditions

Measurement views: Frequency and Phase Settling

Group

Sense commands

Syntax

```
[SENSe]:{FSETtling|PSETtling}:MAXTracepoints { ONEK | TENK | HUNDredk | NEVerdecimate }
[SENSe]:{FSETtling|PSETtling}:MAXTracepoints?
```

Arguments

`ONEK` sets the maximum trace points to 1 k.

TENK sets the maximum trace points to 10 k.

HUNDredk sets the maximum trace points to 100 k.

NEVerdecimate never decimates the trace points.

Examples

SENSe:PSETTLING:MAXTRACEPOINTS TENK sets the maximum trace points to 10 k.

[SENSe]:{FSETtling|PSETtling}:SDURation:MINimum

Sets or queries the minimum settled duration in seconds. This duration time determines the result of :FETCh:{FSETtling | PSETtling}:SLMSd[:PASS]?.

Conditions

Measurement views: Frequency and Phase Settling Time

Group

Sense commands

Syntax

```
[SENSe]:{FSETtling|PSETtling}:SDURation:MINimum <value>
[SENSe]:{FSETtling|PSETtling}:SDURation:MINimum?
```

Arguments

<value>::<NRf> specifies the minimum settled duration in seconds.

Examples

SENSe:FSETTLING:SDURATION:MINIMUM 17.73E-3 sets the minimum settled duration to 17.73 ms.

[SENSe]:{FSETtling|PSETtling}:TARGet:REFerence

Sets or queries the target reference frequency.

Conditions

Measurement views: Frequency and Phase Settling Time

Group

Sense commands

Syntax

```
[SENSe]:{FSETtling|PSETtling}:TARGet:REFerence { AUTO | MFReq }
[SENSe]:{FSETtling|PSETtling}:TARGet:REFerence?
```

Arguments

AUTO causes the instrument to determine the target reference value by averaging at least the last 256 samples at the end of the measurement period.

MFReq specifies that the Measurement Frequency value is used as the target reference frequency.

Default: AUTO

Examples

SENSe:FSETTLING:TARGET:REFERENCE MFREQ specifies that the target reference frequency is the measurement frequency.

[SENSe]:{FSETtling|PSETtling}:TOLerance

Sets or queries the frequency or phase range into which the signal must settle.

Conditions

Measurement views: Frequency and Phase Settling Time

Group

Sense commands

Syntax

```
[SENSe]:{FSETtling|PSETtling}:TOLerance <value>
[SENSe]:{FSETtling|PSETtling}:TOLerance?
```

Arguments

<value>::<NRf> specifies the frequency or phase range.

For Frequency Settling Time:

- Range: 1 Hz to 1 GHz
- Default: 500 kHz

For Phase Settling Time:

- Range: 0.1 degree to 180 degrees
- Default: 1 degree

Examples

SENSe:FSETTLING:TOLERANCE 2.5E5 sets the tolerance to 250 kHz.

[SENSe]:FVTime[:MEASview<y>]:CLEar:RESults (No Query Form)

Restarts multi-trace functions (Average and Max/Min Hold).

Conditions

Measurement views: Frequency versus Time

Group

Sense commands

Syntax

```
[SENSe]:FVTime[:MEASview<y>]:CLEar:RESults
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Examples

SENSe:FVTIME:MEASview1:CLEAR:RESULTS restarts multi-trace functions.

[SENSe]:FVTime[:MEASview<y>]:COLor

Sets or queries the color palette of three-dimensional graphs.

Conditions

Measurement views: Frequency versus Time.

Group

Sense commands

Syntax

```
[SENSe]:FVTime[:MEASview<y>]:COLor {RED | GREen | BLUe | CYAN | BCYan | YELLow | MAGenta
| GRAY | TEMPerature | SPECtral}
```

```
[SENSe]:FVTime[:MEASview<y>]:COLor?
```

Arguments

The following table lists the arguments.

Argument	Description
RED	Red
GREen	Green
BLUe	Blue
CYAN	Cyan
BCYan	Binary cyan
YELLow	Yellow
MAGenta	Magenta
GRAY	Gray
TEMPerature	Temperature
SPECtral	Spectral

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`SENSE:FVTIME:COLOR TEMPERATURE` selects the temperature color palette.

[SENSe]:FVTime[:MEASview<y>]:COLor:MAXimum

Sets or queries the maximum value of the color axis in the Frequency versus Time measurement.

Conditions

Measurement views: Frequency versus Time.

Group

Sense commands

Syntax

`[SENSe]:FVTime[:MEASview<y>]:COLor:MAXimum <value>`

`[SENSe]:FVTime[:MEASview<y>]:COLor:MAXimum?`

Arguments

`<value> =::<NRf>` specifies the maximum value of the color axis.

Range: The minimum value to 100%.

The minimum value is set using the `[SENSe]:FVTime[:MEASview<y>]:COLor:MINimum` command.

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`SENSE:FVTIME:COLOR:MAXIMUM 90` sets the maximum value of the color axis to 90%.

[SENSe]:FVTime[:MEASview<y>]:COLor:MINimum

Sets or queries the minimum value of the color axis in the Frequency versus Time measurement.

Conditions

Measurement views: Frequency versus Time

Group

Sense commands

Syntax

`[SENSe]:FVTime[:MEASview<y>]:COLor:MINimum <value>`

`[SENSe]:FVTime[:MEASview<y>]:COLor:MINimum?`

Arguments

`<value>=<NRf>` specifies the minimum value of the color axis.

Range: 0% to the maximum value.

The maximum value is set using the `[SENSe]:FVTime[:MEASview<y>]:COLor:MAXimum` command.

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`SENSE:FVTIME:COLOR:MINIMUM 10` sets the minimum value of the color axis to 10%.

[SENSe]:FVTime[:MEASview<y>]:COLor:REVerse[:STATe]

Determines whether to reverse the color palette in the Frequency versus Time view. When reversed, the highest density takes the color normally used for the lowest.

Conditions

Measurement views: Frequency versus Time.

Group

Sense commands

Syntax

`[SENSe]:FVTime[:MEASview<y>]:COLor:REVerse[:STATe] {OFF | ON | 0 | 1}`

`[SENSe]:FVTime[:MEASview<y>]:COLor:REVerse[:STATe]?`

Arguments

`OFF` or `0` uses the color palette in its normal order.

`ON` or `1` reverses the color palette.

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`SENSE:FVTIME:COLOR:REVERSE:STATE ON` reverses the color palette.

[SENSe]:FVTime[:MEASview<y>]:FREQuency:CENTer

Sets or queries the center frequency in the Frequency versus Time measurement.

NOTE: The center, start and stop frequencies are set interlocking each other with the following relationships: (start frequency) = (center frequency) - (span)/2 and (stop frequency) = (center frequency) + (span)/2.

Conditions

Measurement views: Frequency versus Time

Group

Sense commands

Syntax

```
[SENSe]:FVTime[:MEASview<y>]:FREQuency:CENTer <value>
[SENSe]:FVTime[:MEASview<y>]:FREQuency:CENTer?
```

Related Commands

[SENSe]:FVTime[:MEASview<y>]:FREQuency:STARt (on page 2136)

[SENSe]:FVTime:MEASview<y>:FREQuency:STOP (on page 2137)

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

`<value>::<NRf>` specifies the center frequency.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples

`SENSe:FVTIME:MEASview1:FREQUENCY:CENTER 7.5GHz` sets the center frequency to 7.5 GHz.

[SENSe]:FVTime[:MEASview<y>]:FREQuency:SPAN

Sets or queries the frequency span in the Frequency versus Time measurement.

Conditions

Measurement views: Frequency versus Time

Group

Sense commands

Syntax

```
[SENSe]:FVTime[:MEASview<y>]:FREQuency:SPAN <value>
[SENSe]:FVTime[:MEASview<y>]:FREQuency:SPAN?
```

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

`<value>::<NRf>` is the frequency span.

Range: 10 Hz to 40 MHz (Standard) / 110MHz (Option 110)

Examples

`SENSe:FVTIME:MEASview1:FREQUENCY:SPAN 20MHz` sets the span to 20 MHz.

[SENSe]:FVTime[:MEASview<y>]:FREQuency:SPAN:LINK

Sets or queries if frequency span is linked or not in the Frequency versus Time measurement.

Conditions

Measurement views: Frequency versus Time

Group

Sense commands

Syntax

```
[SENSe]:FVTime[:MEASview<y>]:FREQuency:SPAN:LINK <value>
[SENSe]:FVTime[:MEASview<y>]:FREQuency:SPAN:LINK?
```

Arguments

`OFF` or `0` specifies that Link to Span is not checked.

`ON` or `1` specifies that Link to Span is checked .

Examples

`SENSe:FVTime:MEASview1:FREQuency:SPAN:LINK?` might return 0 or 1 based on condition of Link to Span checkbox.

[SENSe]:FVTime[:MEASview<y>]:FREQuency:STARt

Sets or queries the measurement start frequency (left edge on the graph) in the Frequency versus Time measurement.

The center, start and stop frequencies are set interlocking each other. Refer to the `[SENSe]:FVTime[:MEASview<y>]:FREQuency:CENTer` command.

Conditions

Measurement views: Frequency versus Time

Group

Sense commands

Syntax

```
[SENSe]:FVTime[:MEASview<y>]:FREQuency:STARt <value>
[SENSe]:FVTime[:MEASview<y>]:FREQuency:STARt?
```

Related Commands

[SENSe]:FVTime[:MEASview<y>]:FREQuency:STOP (on page 2137)

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

`<value>::<NRf>` is the measurement start frequency.

Range: (center frequency) ± (span)/2.

Examples

`SENSe:FVTIME:MEASview1:FREQUENCY:START 6.95GHz` sets the start frequency to 6.95 GHz.

[SENSe]:FVTime[:MEASview<y>]:FREQuency:STEP

Sets or queries the frequency step size (the amount per press by which the up or down key changes the setting value). Programming a specified step size sets `[SENSe]:FVTime[:MEASview<y>]:FREQuency:STEP:AUTO OFF`.

Conditions

Measurement views: Frequency versus Time

Group

Sense commands

Syntax

```
[SENSe]:FVTime[:MEASview<y>]:FREQuency:STEP <value>
[SENSe]:FVTime[:MEASview<y>]:FREQuency:STEP?
```

Related Commands

[SENSe]:FVTime[:MEASview<y>]:FREQuency:STEP:AUTO (on page 2137)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<value>::<NRf> specifies the frequency step size.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples

SENSe:FVTIME:MEASview1:FREQUENCY:STEP 1.5kHz sets the step size to 1.5 kHz.

[SENSe]:FVTime[:MEASview<y>]:FREQuency:STEP:AUTO

Determines whether to set the frequency step size automatically or manually.

Conditions

Measurement views: Frequency versus Time

Group

Sense commands

Syntax

```
[SENSe]:FVTime[:MEASview<y>]:FREQuency:STEP:AUTO { OFF | ON | 0 | 1 }
[SENSe]:FVTime[:MEASview<y>]:FREQuency:STEP:AUTO?
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

OFF or 0 specifies that the frequency step size is set manually using the [SENSe]:FVTime[:MEASview<y>]:FREQuency:STEP command.

ON or 1 specifies that the frequency step size is set automatically.

Examples

SENSe:FVTIME:MEASview1:BANDWIDTH:AUTO ON sets the frequency step size automatically.

[SENSe]:FVTime[:MEASview<y>]:FREQuency:STOP

Sets or queries the measurement stop frequency (right edge of the graph) in the Frequency versus Time measurement.

The center, start and stop frequencies are set interlocking each other. Refer to the [SENSe]:FVTime[:MEASview<y>]:FREQuency:CENTer command.

Conditions

Measurement views: Frequency versus Time

Group

Sense commands

Syntax

```
[SENSe]:FVTime[:MEASview<y>]:FREQuency:STOP <value>
[SENSe]:FVTime[:MEASview<y>]:FREQuency:STOP?
```

Related Commands

[SENSe]:FVTime[:MEASview<y>]:FREQuency:STARt (on page 2136)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<value>::<NRf> is the measurement stop frequency.

Range: (center frequency) ± (span)/2.

Examples

SENSe:FVTIME:MEASview1:FREQUENCY:STOP 7.05GHz sets the stop frequency to 7.05 GHz.

[SENSe]:FVTime[:MEASview<y>]:MAXTracepoints

Sets or queries the maximum trace points in the Frequency versus Time measurement.

Conditions

Measurement views: Frequency versus Time

Group

Sense commands

Syntax

```
[SENSe]:FVTime[:MEASview<y>]:MAXTracepoints { ONEK | TENK | HUNDredk | NEVerdecimate }
[SENSe]:FVTime[:MEASview<y>]:MAXTracepoints?
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

ONEK sets the maximum trace points to 1 k.

TENK sets the maximum trace points to 10 k.

HUNDredk sets the maximum trace points to 100 k.

NEVerdecimate never decimates the trace points.

Examples

SENSe:FVTIME:MEASview1:MAXTRACEPOINTS TENK sets the maximum trace points to 10 k.

[SENSe]:FVTime[:MEASview<y>]:OGRAm:COLor

Sets or queries the ogram color scheme.

Conditions

Applicable measurement view must be active.

Group

Sense commands

Syntax

```
[SENSe]:FVTime[:MEASview<y>]:OGRAm:COLor {RED | GREen | BLUe | CYAN | BCYan | YELLow | MAGenta | GRAY | TEMPerature | SPECtral}
```

```
[SENSe]:AVTime[:MEASview<y>]:OGRAm:COLor?
```

Related Commands

[SENSe]:FVTime[:MEASview<y>]:OGRAm:COLor:MAXimum (on page 2139)

Arguments

The following table lists the arguments.

Argument	Description
RED	Red
GREen	Green
BLUe	Blue
CYAN	Cyan
BCYan	Binary cyan
YELLow	Yellow
MAGenta	Magenta
GRAY	Gray
TEMPerature	Temperature
SPECtral	Spectral

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Returns

```
{RED | GREen | BLUe | CYAN | BCYan | YELLow | MAGenta | GRAY | TEMPerature | SPECtral}
```

Examples

`SENSe:FVTime:MEASview1:OGRAm:COLor TEMPerature` selects the temperature color palette.

[SENSe]:FVTime[:MEASview<y>]:OGRAm:COLor:MAXimum

Sets or queries the maximum value of the color axis in the ogram.

Conditions

Applicable measurement view must be active.

Group

Sense commands

Syntax

`[SENSe]:FVTime[:MEASview<y>]:OGRAm:COLor:MAXimum <value>`

`[SENSe]:FVTime[:MEASview<y>]:OGRAm:COLor:MAXimum?`

Related Commands

[SENSe]:FVTime[:MEASview<y>]:OGRAm:COLor:MINimum (on page 2140)

Arguments

`<value>::=<NRf>` specifies the maximum value of the color axis.

Range: -100 to +100 dBm.

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Returns

`<NRf>` specifies the maximum value of the color axis.

Examples

`SENSe:FVTime:MEASview1:OGRAm:COLor:MAXimum 10` sets the maximum value of the color axis to 10 dBm.

[SENSe]:FVTime[:MEASview<y>]:OGRAm:COLor:MINimum

Sets or queries the minimum value of the color axis in the ogram.

Conditions

Applicable measurement view must be active.

Group

Sense commands

Syntax

`[SENSe]:FVTime[:MEASview<y>]:OGRAm:COLor:MINimum <value>`

`[SENSe]:FVTime[:MEASview<y>]:OGRAm:COLor:MINimum?`

Related Commands

[SENSe]:FVTime[:MEASview<y>]:OGRAm:COLor:MAXimum (on page 2139)

Arguments

`<value>::=<NRf>` specifies the minimum value of the color axis.

Range: -100 to +100 dBm.

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional. By default y is 1.

Returns

<NRf> specifies the minimum value of the color axis.

Examples

SENSe:FVTime:MEASview1:OGRAm:COLor:MINimum 10 sets the minimum value of the color axis to 10 dBm.

[SENSe]:FVTime[:MEASview<y>]:OGRAm:COLor:REVerse

Sets or queries reverse color scheme state.

Conditions

Applicable measurement view must be active.

Group

Sense commands

Syntax

[SENSe]:FVTime[:MEASview<y>]:OGRAm:COLor:REVerse { OFF | ON | 0 | 1 }

[SENSe]:FVTime[:MEASview<y>]:OGRAm:COLor:REVerse?

Related Commands

[SENSe]:FVTime[:MEASview<y>]:OGRAm:COLor (on page 2138)

Arguments

OFF or 0 disables Reverse color scheme.

ON or 1 enables Reverse color scheme.

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional. By default y is 1.

Returns

0 disables Reverse color scheme.

1 enables Reverse color scheme.

Examples

SENSe:FVTime:MEASview1:OGRAm:COLor:REVerse OFF disables reverse color scheme.

[SENSe]:FVTime[:MEASview<y>]:PLOT

Sets or queries the Frequency vs time display mode.

Conditions

Applicable measurement view must be active.

Group

Sense commands

Syntax

[SENSe]:FVTime[:MEASview<y>]:PLOT {TRACE | OGRAM |SPLIT}

```
[SENSe]: FVTime [:MEASview<y>]:PLOT?
```

Arguments

`TRACE` displays time plot .

`OGRAM` displays Ogram plot.

`SPLIT` displays both ogram and time plots.

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Returns

```
{TRACE | OGRAM | SPLIT}
```

Examples

`SENSe:FVTime:MEASview1:PLOT TRACE` sets plot type to TRACE (Time plot).

[SENSe]:FVTime[:MEASview<y>]:SELEcted:SOURce

Sets or queries the Source with respect to the Frequency versus Time Measurement View nuumber

Conditions

Measurement views: Frequency versus Time with specified Measurement view number

Source cannot be changed when acquisition or analysis is in progress. Stop the acquisition and change the Source.

Group

Sense commands

Syntax

```
SENSe:FVTime[:MEASview<y>]:SELEcted:SOURce {"Source 1" | "Source 2" | "Source 3" | "Source 4"}
SENSe:FVTime[:MEASview<y>]:SELEcted:SOURce?
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Examples

`SENSe:FVTime:MEASview1:SELEcted:SOURce?` queries or sets the Source for Frequency versus Time Measurement View 1.

[SENSe]:IQVTime[:MEASview<y>]:CLEar:RESults (No Query Form)

Restarts multi-trace functions (Average and Max/Min Hold).

Conditions

Measurement views: RF I&Q versus Time

Group

Sense commands

Syntax

`[SENSe]:IQVTime[:MEASview<y>]:CLEar:RESults`

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Examples

`SENSe:IQVTIME:MEASview1:CLEAR:RESULTS` restarts multi-trace functions.

[SENSe]:IQVTime[:MEASview<y>]:FREQuency:CENTer

Sets or queries the center frequency in the RF I&Q versus Time measurement.

NOTE: The center, start and stop frequencies are set interlocking each other with the following relationships: (start frequency) = (center frequency) - (span)/2 and (stop frequency) = (center frequency) + (span)/2.

Conditions

Measurement views: RF I&Q versus Time

Group

Sense commands

Syntax

```
[SENSe]:IQVTime[:MEASview<y>]:FREQuency:CENTer <value>
[SENSe]:IQVTime[:MEASview<y>]:FREQuency:CENTer?
```

Related Commands

[SENSe]:IQVTime[:MEASview<y>]:FREQuency:STARt (on page 2144)

[SENSe]:IQVTime:MEASview<y>:FREQuency:STOP (on page 2146)

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

`<value>::<NRf>` specifies the center frequency.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples

`SENSe:IQVTIME:MEASview1:FREQUENCY:CENTER 7.5GHz` sets the center frequency to 7.5 GHz.

[SENSe]:IQVTime[:MEASview<y>]:FREQuency:SPAN

Sets or queries the frequency span in the RF I&Q versus Time measurement.

Conditions

Measurement views: RF I&Q versus Time

Group

Sense commands

Syntax

```
[SENSe]:IQVTime[:MEASview<y>]:FREQuency:SPAN <value>
[SENSe]:IQVTime[:MEASview<y>]:FREQuency:SPAN?
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<value>::<NRf> is the frequency span.

Range: 10 Hz to 40 MHz (Standard) / 110 MHz (Option 110)

Examples

SENSe:IQVTIME:MEASview1:FREQUENCY:SPAN 20MHz sets the span to 20 MHz.

[SENSe]:IQVTime[:MEASview<y>]:FREQuency:SPAN:LINK

Sets or queries if frequency span is linked or not in the RF I & Q versus Time measurement.

Conditions

Measurement views: RF I & Q versus Time

Group

Sense commands

Syntax

```
[SENSe]:IQVTime[:MEASview<y>]:FREQuency:SPAN:LINK <value>
[SENSe]:IQVTime[:MEASview<y>]:FREQuency:SPAN:LINK?
```

Arguments

OFF or 0 specifies that Link to Span is not checked.

ON or 1 specifies that Link to Span is checked .

Examples

SENSe:IQVTime:MEASview1:FREQuency:SPAN:LINK? might return 0 or 1 based on condition of Link to Span checkbox.

[SENSe]:IQVTime[:MEASview<y>]:FREQuency:STARt

Sets or queries the measurement start frequency (left edge on the graph) in the RF I&Q versus Time measurement.

The center, start and stop frequencies are set interlocking each other. Refer to the [SENSe]:IQVTime[:MEASview<y>]:FREQuency:CENTer command.

Conditions

Measurement views: RF I&Q versus Time

Group

Sense commands

Syntax

```
[SENSe]:IQVTime[:MEASview<y>]:FREQuency:STARt <value>
[SENSe]:IQVTime[:MEASview<y>]:FREQuency:STARt?
```

Related Commands

[SENSe]:IQVTime[:MEASview<y>]:FREQuency:STOP (on page 2146)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<value>::<NRf> is the measurement start frequency.

Range: (center frequency) ± (span)/2.

Examples

SENSe:IQVTIME:MEASview1:FREQUENCY:START 6.95GHz sets the start frequency to 6.95 GHz.

[SENSe]:IQVTime[:MEASview<y>]:FREQuency:STEP

Sets or queries the frequency step size (the amount per press by which the up or down key changes the setting value). Programming a specified step size sets [SENSe]:IQVTime[:MEASview<y>]:FREQuency:STEP:AUTO OFF.

Conditions

Measurement views: RF I&Q versus Time

Group

Sense commands

Syntax

```
[SENSe]:IQVTime[:MEASview<y>]:FREQuency:STEP <value>
[SENSe]:IQVTime[:MEASview<y>]:FREQuency:STEP?
```

Related Commands

[SENSe]:IQVTime[:MEASview<y>]:FREQuency:STEP:AUTO (on page 2146)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<value>::<NRf> specifies the frequency step size.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples

SENSe:IQVTIME:MEASview1:FREQUENCY:STEP 1.5kHz sets the step size to 1.5 kHz.

[SENSe]:IQVTime[:MEASview<y>]:FREQuency:STEP:AUTO

Determines whether to set the frequency step size automatically or manually.

Conditions

Measurement views: RF I&Q versus Time

Group

Sense commands

Syntax

```
[SENSe]:IQVTime[:MEASview<y>]:FREQuency:STEP:AUTO { OFF | ON | 0 | 1 }
[SENSe]:IQVTime[:MEASview<y>]:FREQuency:STEP:AUTO?
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

OFF or 0 specifies that the frequency step size is set manually using the [SENSe]:IQVTime[:MEASview<y>]:FREQuency:STEP command.

ON or 1 specifies that the frequency step size is set automatically.

Examples

SENSe:IQVTIME:MEASview1:FREQUENCY:STEP:AUTO ON sets the frequency step size automatically.

[SENSe]:IQVTime[:MEASview<y>]:FREQuency:STOP

Sets or queries the measurement stop frequency (right edge of the graph) in the RF I&Q versus Time measurement.

The center, start and stop frequencies are set interlocking each other. Refer to the [SENSe]:IQVTime[:MEASview<y>]:FREQuency:CENTer command.

Conditions

Measurement views: RF I&Q versus Time

Group

Sense commands

Syntax

```
[SENSe]:IQVTime[:MEASview<y>]:FREQuency:STOP <value>
[SENSe]:IQVTime[:MEASview<y>]:FREQuency:STOP?
```

Related Commands

[SENSe]:IQVTime[:MEASview<y>]:FREQuency:STARt (on page 2144)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<value>::<NRf> is the measurement stop frequency.

Range: (center frequency) ± (span)/2.

Examples

`SENSe:IQVTIME:MEASview1:FREQUENCY:STOP 7.05GHz` sets the stop frequency to 7.05 GHz.

[SENSe]:IQVTime[:MEASview<y>]:MAXTracepoints

Sets or queries the maximum trace points in the RF I&Q versus Time measurement.

Conditions

Measurement views: RF I&Q versus Time

Group

Sense commands

Syntax

```
[SENSe]:IQVTime[:MEASview<y>]:MAXTracepoints { ONEK | TENK | HUNDredk | NEVerdecimate }
[SENSe]:IQVTime[:MEASview<y>]:MAXTracepoints?
```

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

`ONEK` sets the maximum trace points to 1 k.

`TENK` sets the maximum trace points to 10 k.

`HUNDredk` sets the maximum trace points to 100 k.

`NEVerdecimate` never decimates the trace points.

Examples

`SENSe:IQVTIME:MEASview1:MAXTRACEPOINTS TENK` sets the maximum trace points to 10 k.

[SENSe]:IQVTime[:MEASview<y>]:SELEcted:SOURce

Sets or queries the Source with respect to the RF I&Q versus Time Measurement View nuumber

Conditions

Measurement views: RF I&Q versus Time

Source cannot be changed when acquisition or analysis is in progress. Stop the acquisition and change the Source.

Group

Sense commands

Syntax

```
SENSe:IQVTime[:MEASview<y>]:SELEcted:SOURce {"Source 1" | "Source 2" | "Source 3" | "Source 4"}
SENSe:IQVTime[:MEASview<y>]:SELEcted:SOURce?
```

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Examples

`SENSe:IQVTime:MEASview1:SELEcted:SOURce?` queries or sets the Source for RF I&Q versus Time.

SENSe:LTE:ACLR:BANDwidth:RESolution

Sets or queries the resolution bandwidth (RBW) in the LTE ACLR display.

Conditions

Measurement views: LTE ACLR

Group

Sense commands

Syntax

`SENSe:LTE:ACLR:BANDwidth:RESolution`

Arguments

`<value>::<NRf>` specifies the resolution bandwidth

Returns

<NR3>

Examples

`SENSe:LTE:ACLR:BANDWIDTH:RESOLUTION?` might return `100.0000000E+3`, indicating the RBW is 100 kHz.

SENSe:LTE:ACLR:BANDwidth:VIDeo

Sets or queries the video bandwidth (VBW) in the LTE ACLR display.

Conditions

Measurement views: LTE ACLR

Group

Sense commands

Syntax

`SENSe:LTE:ACLR:BANDwidth:VIDeo <value>`

Arguments

`<value>::<NRf>` specifies the video bandwidth

Returns

<NR3>

Examples

`SENSe:LTE:ACLR:BANDWIDTH:VIDEO 2.0000000E+3` sets the VBW to 2 kHz.

SENSe:LTE:ACLR:BANDwidth:VIDeo:STATe

Sets or queries enabling or disabling the video bandwidth (VBW) in the LTE ACLR display.

Conditions

Measurement views: LTE ACLR

Group

Sense commands

Syntax

```
SENSe:LTE:ACLR:BANDwidth:VIDeo:STATe { 0 | 1 }
SENSe:LTE:ACLR:BANDwidth:VIDeo:STATe?
```

Arguments

`0` disables the VBW.

`1` enables the VBW.

Returns

0 means that the VBW is disabled.

1 means that the VBW is enabled.

Examples

`SENSe:LTE:ACLR:BANDWIDTH:VIDEO:STATE 1` sets the VBW to automatic (enabled) and is decided by the analysis module.

SENSe:LTE:ACLR:CHANnel:IBANdwidth

Sets or queries the reference channel integration bandwidth for the LTE ACLR display.

Conditions

Measurement views: LTE ACLR

Group

Sense commands

Syntax

```
SENSe:LTE:ACLR:CHANnel:IBANdwidth <value>
SENSe:LTE:ACLR:CHANnel:IBANdwidth?
```

Arguments

`<value>::<NRf>` specifies the channel integration bandwidth.

Returns

<NR3>

Examples

`SENSe:LTE:ACLR:CHANNEL:IBANDWIDTH?` might return `18.000000E+6`, indicating that the integration bandwidth is 18 MHz.

SENSe:LTE:ACLR:CHANnel:WIDTh

Sets or queries the reference channel width for the LTE ACLR display.

Conditions

Measurement views: LTE ACLR

Group

Sense commands

Syntax

```
SENSe:LTE:ACLR:CHANnel:WIDTh <value>
SENSe:LTE:ACLR:CHANnel:WIDTh?
```

Arguments

`<value>::<NRf>` specifies the reference channel width.

Returns

<NR3>

Examples

`SENSe:LTE:ACLR:CHANNEL:WIDTH 5.00000000E+6` sets the reference channel width to 5 MHz.

SENSe:LTE:ACLR:CLEar:RESults (No Query Form)

Clears the results in the LTE ACLR measurement.

Conditions

Measurement views: LTE ACLR

Group

Sense commands

Syntax

```
SENSe:LTE:ACLR:CLEar:RESults
```

Arguments

None

Examples

`SENSe:LTE:ACLR:CLEAR:RESULTS` clears the results of the LTE ACLR measurement.

SENSe:LTE:ACLR:FREQuency

Sets or queries the measurement center frequency in the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

Group

Sense commands

Syntax

```
SENSe:LTE:ACLR:FREQuency <value>
SENSe:LTE:ACLR:FREQuency?
```

Arguments

<value>::<NRf> specifies the center frequency

Returns

<NR3>

Examples

SENSe:LTE:ACLR:FREQUENCY 1.000000000E+9 sets the center frequency to 1.0 GHz.

SENSe:LTE:ACLR:MFILter

Sets or queries the shape of the filter for the LTE ACLR measurement.

Conditions

Measurement view: LTE ACLR

Group

Sense commands

Syntax

```
SENSe:LTE:ACLR:MFILter { GAUSsian | RECTangular }
SENSe:LTE:ACLR:MFILter?
```

Arguments

GAUSsian: sets the filter shape to Gaussian.

RECTangular: sets the filter shape to rectangular.

Returns

GAUSSIAN: means the filter shape is Gaussian.

RECTangular: means the filter shape is rectangular.

Examples

SENSe:LTE:ACLR:MFILTER RECT sets the filter shape to rectangular.

SENSe:LTE:ACLR:MODE:REALtime

Sets or queries the real-time LTE ACLR measurement mode. When set to real-time, the reference channel and all requested adjacent channel regions are acquired together. When real-time if off, the measurement is done in Swept mode (an acquisition for each adjacent band).

Conditions

Measurement view: LTE ACLR

Group

Sense commands

Syntax

```
SENSe:LTE:ACLR:MODE:REALtime { OFF | ON | 0 | 1 }
SENSe:LTE:ACLR:MODE:REALtime?
```

Arguments

`OFF` or `0` turns off the automatic update.

`ON` or `1` turns on automatic updates center frequency.

Returns

OFF or 0 means automatic update is off.

ON or 1 means automatic updates are off.

Examples

`SENSe:LTE:ACLR:MODE:REALTIME?` might return `1` to indicate the real-time checkbox in the LTE ACLR control panel is enabled.

SENSe:LTE:ACLR:NFLoor:ACQuire (No Query Form)

Enables measurement of the noise floor for the LTE ACLR measurement.

Conditions

Measurement view: LTE ACLR

Group

Sense commands

Syntax

`SENSe:LTE:ACLR:NFLoor:ACQuire`

Arguments

None

Examples

`SENSe:LTE:ACLR:NFLOOR:ACQUIRE` will enable measurement of the noise floor in the LTE ACLR measurement.

SENSe:LTE:ACLR:NFLoor:CORRected? (Query Only)

Returns whether or not noise correction was applied on the last acquisition in the LTE ACLR measurement.

Conditions

Measurement view: LTE ACLR

Group

Sense commands

Syntax

`SENSe:LTE:ACLR:NFLoor:CORRected?`

Returns

1 means that noise correction was applied on the last acquisition.

0 means that noise correction was not applied on the last acquisition.

Examples

SENSe:LTE:ACLR:NFLOOR:CORRECTED? might return 1, indicating that noise correction was applied on the last acquisition.

SENSe:LTE:ACLR:NFLoor:STATe

Determines whether to enable or disable the correction for noise floor for the LTE ACLR measurement.

Conditions

Measurement view: LTE ACLR

Group

Sense commands

Syntax

```
SENSe:LTE:ACLR:NFLoor:STATe { OFF | ON | 0 | 1 }
SENSe:LTE:ACLR:NFLoor:STATe?
```

Arguments

OFF or 0 disables noise correction in the measurement view.

ON or 1 enables noise correction in the measurement view.

Returns

OFF or 0 means that noise correction is disabled in the measurement view.

ON or 1 means that noise correction is enabled in the measurement view.

Examples

SENSe:LTE:ACLR:NFLOOR:STATE OFF disables noise correction in the measurement view.

SENSe:LTE:ACLR:OFFSet<x>:BANDwidth:RESolution

Sets or queries the resolution bandwidth (RBW) value in the specified row in the Offsets & Limits table for the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

The parameter <x> represents the row in the Offsets & Limits table.

Group

Sense commands

Syntax

```
SENSe:LTE:ACLR:OFFSet<x>:BANDwidth:RESolution <value>
SENSe:LTE:ACLR:OFFSet<x>:BANDwidth:RESolution?
```

Arguments

<value>::<NRf> specifies the RBW in the specified row.

Returns

<NR3>

Examples

`SENSe:LTE:ACLR:OFFSET4:BANDWIDTH:RESOLUTION 30.000000000E+3` sets the RBW to 30 kHz for row 4.

SENSe:LTE:ACLR:OFFSet<x>:BANDwidth:RESolution:COUNt

Sets or queries the count for the resolution bandwidths in the specified row in the Offsets & Limits table for the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

The parameter <x> represents the row in the Offsets & Limits table.

Group

Sense commands

Syntax

```
SENSe:LTE:ACLR:OFFSet<x>:BANDwidth:RESolution:COUNt <value>
SENSe:LTE:ACLR:OFFSet<x>:BANDwidth:RESolution:COUNt?
```

Arguments

`<value>::<NR1>` specifies a count integer

Returns

<NR1>

Examples

`SENSe:LTE:ACLR:OFFSET1:BANDWIDTH:RESOLUTION:COUNT 10` sets the count for row 1 to 10.

SENSe:LTE:ACLR:OFFSet<X>:BANDwidth:RESolution:FILTer

Sets or queries the filter bandwidth resolution under Offset & Limit table tab for the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

Parameter <x> represents the row in the Offsets & Limits table

Group

Sense commands

Syntax

```
SENSe:LTE:ACLR:OFFSet<X>:BANDwidth:RESolution:FILTer <value>
SENSe:LTE:ACLR:OFFSet<X>:BANDwidth:RESolution:FILTer?
```

Arguments

`<value>::<NRf>` specifies the filter bandwidth resolution in the Offset & Limit table.

Returns

<NR3>

Examples

`SENSe:LTE:ACLR:OFFSET1:BANDWIDTH:RESOLUTION:FILTER?` might return `100.0000000000E+3` indicating the value of filter bandwidth of the first row is 100 kHz.

SENSe:LTE:ACLR:OFFSet<x>:BANDwidth:RESolution:VIDeo

Sets or queries the resolution video bandwidth (VBW) value in the specified row in the Offsets & Limits table for the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

The parameter <x> represents the row in the Offsets & Limits table.

Group

Sense commands

Syntax

```
SENSe:LTE:ACLR:OFFSet<x>:BANDwidth:RESolution:VIDeo <value>
SENSe:LTE:ACLR:OFFSet<x>:BANDwidth:RESolution:VIDeo?
```

Arguments

`<value>::<NRf>` specifies the VBW in the specifies row.

Returns

<NR3>

Examples

`SENSe:LTE:ACLR:OFFSET1:BANDWIDTH:RESOLUTION:VIDEO 40.000000000E+3` sets the resolution VBW to 40 kHz for row 1.

SENSe:LTE:ACLR:OFFSet<x>:BANDwidth:RESolution:VIDeo:STATe

Sets or queries whether to enable or disable the video resolution bandwidth (VBW) in the specified row in the Offsets & Limits table for the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

The parameter <x> represents the row in the Offsets & Limits table.

Group

Sense commands

Syntax

```
SENSe:LTE:ACLR:OFFSet<x>:BANDwidth:RESolution:VIDeo:STATe{ 0 | 1 }
SENSe:LTE:ACLR:OFFSet<x>:BANDwidth:RESolution:VIDeo:STATe?
```

Arguments

`0` disables the VBW.

`1` enables the VBW.

Returns

0 means the resolution VBW is disabled in the specified row.

1 means the VBW is enabled in the specified row.

Examples

`SENSe:LTE:ACLR:OFFSET1:BANDWIDTH:RESOLUTION:VIDEO:STATE 1` enables the resolution VBW for row 1.

SENSe:LTE:ACLR:OFFSet<x>:DETection

Sets or queries the detection settings for all offsets for the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

Group

Sense commands

Syntax

```
SENSe:LTE:ACLR:OFFSet<x>:DETection { AVERage | PEAK }
SENSe:LTE:ACLR:OFFSet<x>:DETection?
```

Arguments

`AVERage`: Sets the offsets setting Average (VRMS)

`PEAK`: Sets the offsets setting is the positive peak

Returns

AVERage means the offsets setting Average (VRMS)

PEAK means the offsets setting is the positive peak

Examples

`SENSe:LTE:ACLR:OFFSET<x>:DETECTION AVER` sets the detection settings for all offsets to Average RMS.

SENSe:LTE:ACLR:OFFSet<x>:FREQuency:BANDwidth

Sets or queries the frequency bandwidth value for the specified row of the Offsets & Limits table for the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

The parameter <x> represents the row in the Offsets & Limits table.

Group

Sense commands

Syntax

```
SENSe:LTE:ACLR:OFFSet<x>:FREQuency:BANDwidth <value>
SENSe:LTE:ACLR:OFFSet<x>:FREQuency:BANDwidth?
```

Arguments

`<value>::<NRf>` specifies the frequency bandwidth value

Returns

<NR3>

Examples

`SENSE:LTE:ACLR:OFFSET1:FREQUENCY:BANDWIDTH?` might return `4.5150000000E+6`, indicating frequency bandwidth value of row 1 in the Offset & Limits table.

SENSe:LTE:ACLR:OFFSet<x>:FREQuency:OFFSet

Sets or queries the frequency offset value of the specified row in the Offsets & Limits table for the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

The parameter <x> represents the row in the Offsets & Limits table.

Group

Sense commands

Syntax

```
SENSe:LTE:ACLR:OFFSet<x>:FREQuency:OFFSet <value>
SENSe:LTE:ACLR:OFFSet<x>:FREQuency:OFFSet?
```

Arguments

`<value>::<NRf>` specifies the frequency offset value

Returns

<NR3>

Examples

`SENSe:LTE:ACLR:OFFSET1:FREQUENCY:OFFSET?` might return `5.0000000000E+6`, indicating frequency offset value of row 1 in the Offset & Limits table tab.

SENSe:LTE:ACLR:OFFSet<x>:LIMit:ABSolute

Sets or queries the absolute limit value of the row in the Offset & Limit table tab for the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

The parameter <x> represents the row in the Offsets & Limits table.

Group

Sense commands

Syntax

```
SENSe:LTE:ACLR:OFFSet<x>:LIMit:ABSolute <value>
SENSe:LTE:ACLR:OFFSet<x>:LIMit:ABSolute?
```

Arguments

`<value>::<NRf>` specifies the absolute limit value of the specified row in the Offset & Limit table.

Returns

<NRf>

Examples

`SENSe:LTE:ACLR:OFFSET1:LIMIT:ABSOLUTE?` might return `-6.4500000000`, indicating the absolute limit value of row 1 is –6.4500000000 dBm.

SENSe:LTE:ACLR:OFFSet<x>:LIMit:MASK

Sets or queries the limit mask setting for the specified row in the Offsets & Limits table for the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

Parameter <x> represents the row in the Offsets & Limits table

Group

Sense commands

Syntax

```
SENSe:LTE:ACLR:OFFSet<x>:LIMit:MASK { ABS | REL | AND | OR }
SENSe:LTE:ACLR:OFFSet<x>:LIMit:MASK?
```

Arguments

`ABS` specifies the limit mask is set to Absolute.

`REL` specifies the limit mask is set to Relative

`AND` specifies the limit mask is set to Absolute AND Relative.

`OR` specifies the limit mask is set to Absolute OR Relative.

Returns

ABS means that the limit mask is set to Absolute.

REL means that the limit mask is set to Relative

AND means that the limit mask is set to Absolute AND Relative.

OR means that the limit mask is set to Absolute OR Relative.

Examples

`SENSe:LTE:ACLR:OFFSET3:LIMIT:MASK REL` sets the limit mask for row 3 to Relative.

SENSe:LTE:ACLR:OFFSet<x>:LIMit:RELative

Sets or queries the relative limit value of channel under Offset & Limit table tab for the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

Parameter <x> represents the row in the Offsets & Limits table

Group

Sense commands

Syntax

```
SENSe:LTE:ACLR:OFFSet<x>:LIMit:RELative <value>
SENSe:LTE:ACLR:OFFSet<x>:LIMit:RELative?
```

Arguments

`<value>::<NRf>` specifies the relative limit value of the specified row in the Offset & Limit table.

Returns

`<NRf>`

Examples

`SENSe:LTE:ACLR:OFFSET1:LIMIT:RELATIVE?` might return `-44.2000000000`, indicating the relative limit value of row 1 is -44.2000000000 dB.

SENSe:LTE:ACLR:OFFSet<x>:LIMit:SIDE

Sets or queries the side of the offset limit for the specified row in the Offsets & Limits table for the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

Parameter <x> represents the row in the Offsets & Limits table

Group

Sense commands

Syntax

```
SENSe:LTE:ACLR:OFFSet<x>:LIMit:SIDE { LEFT | RIGHt | BOTH }
SENSe:LTE:ACLR:OFFSet<x>:LIMit:SIDE?
```

Arguments

`LEFT` specifies the left offset limit.

`RIGHt` specifies the right offset limit.

`BOTH` specifies both sides of the offset limit.

Returns

LEFT means the left side offset limit is used for the specified row. RIGHt means the right side offset limit is used for the specified row. BOTH means both sides of the offset limit are used for the specified row.

Examples

`SENSe:LTE:ACLR:OFFSET1:LIMIT:SIDE BOTH` specifies the measurement uses both sides of the offset in row 1.

SENSe:LTE:ACLR:OFFSet<x>:STATe

Sets or queries the specified row of the Offsets & Limits table for the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

The parameter <x> represents the row in the Offsets & Limits table.

Group

Sense commands

Syntax

```
SENSe:LTE:ACLR:OFFSet<x>:STATe { OFF | ON | 0 | 1 }
SENSe:LTE:ACLR:OFFSet<x>:STATe?
```

Arguments

`OFF` or 0 turns off display of the row in the measurement view. ON or 1 turns on display of the row in the measurement view.

Returns

OFF or 0 means the display of the row in the measurement view is off. ON or 1 means the display of the row in the measurement view is on.

Examples

`SENSe:LTE:ACLR:OFFSET2:STATE OFF` means that the display of row 2 in the Offsets & Limits table.

SENSe:LTE:ACLR:POINts:COUNt

Sets or queries the number of points used for the trace in the LTE ACLR measurement.

Conditions

Measurement view: LTE ACLR

Group

Sense commands

Syntax

```
SENSe:LTE:ACLR:POINts:COUNt <value>
SENSe:LTE:ACLR:POINts:COUNt?
```

Arguments

`<value>::<NRf>` specifies the number of points used for the trace

Returns

`<NR1>`

Examples

`SENSe:LTE:ACLR:POINTS:COUNT?` might return `P8001`, indicating a maximum of 8001 trace points can be used.

SENSe:LTE:ACLR:REFerence:MAGNitude

Sets or queries the reference power magnitude for the LTE ACLR measurement.

Conditions

Measurement view: LTE ACLR

Group

Sense commands

Syntax

```
SENSe:LTE:ACLR:REFerence:MAGNitude <value>
SENSe:LTE:ACLR:REFerence:MAGNitude?
```

Arguments

`<value>::<NRf>` specifies the reference power magnitude

Returns

`<NRf>`

Examples

`SENSe:LTE:ACLR:REFERENCE:MAGNITUDE?` might return a `-51.057`, indicating that the reference power magnitude is −51.057 dBm.

SENSe:LTE:ACLR:REFerence:MAGNitude:AUTO

Sets or queries the reference magnitude to automatic or manual for the LTE ACLR measurement.

Conditions

Measurement view: LTE ACLR

Group

Sense commands

Syntax

```
SENSe:LTE:ACLR:REFerence:MAGNitude:AUTO { 0 | 1 }
SENSe:LTE:ACLR:REFerence:MAGNitude:AUTO?
```

Arguments

`0` sets reference magnitude to Manual.

`1` sets reference magnitude to Automatic.

Returns

0 means the reference magnitude is set to Manual.

1 means the reference magnitude is set to Automatic.

Examples

`SENSe:LTE:ACLR:REFERENCE:MAGNITUDE:AUTO 0` sets the reference magnitude to Manual.

SENSe:LTE:ACLR:REFerence:MAGNitude:DETection

Sets or queries the reference magnitude detection for the LTE ACLR measurement.

Conditions

Measurement view: LTE ACLR

Group

Sense commands

Syntax

```
SENSe:LTE:ACLR:REFerence:MAGNitude:DETection { AVERage | PEAK }
SENSe:LTE:ACLR:REFerence:MAGNitude:DETection?
```

Arguments

`AVER` sets reference magnitude detection to Avg(VRMS) PEAK sets reference magnitude detection to +Peak.

Returns

AVER means the reference magnitude detection is set to Avg(VRMS) PEAK means the reference magnitude detection is set to +Peak.

Examples

`SENSe:LTE:ACLR:REFERENCE:MAGNITUDE:DETECTION AVER` sets the magnitude detection to Avg(RMS).

SENSe:LTE:ACLR:STEP:AUTO

Determines whether to automatically update the center frequency increment/decrement step size in the LTE ACLR display. The step value controls the magnitude of each step when using the front panel controls or arrow key on the keyboard.

Conditions

Measurement view: LTE ACLR

Group

Sense commands

Syntax

```
SENSe:LTE:ACLR:STEP:AUTO { OFF | ON | 0 | 1 }
SENSe:LTE:ACLR:STEP:AUTO?
```

Arguments

`OFF` or `0` turns off the automatic update.

`ON` or `1` turns on automatic updates center frequency.

Returns

OFF or 0 means automatic update is off. ON or 1 means automatic updates are off.

Examples

`SENSe:LTE:ACLR:STEP:AUTO ON` automatically updates the center frequency.

SENSe:LTE:ACLR:STEP:MAGNitude

Sets or queries the step magnitude for the center frequency in the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

Group

Sense commands

Syntax

```
SENSe:LTE:ACLR:STEP:MAGNitude <value>
```

SENSe:LTE:ACLR:STEP:MAGNitude?

Arguments

<value>::<NRf> specifies the center frequency

Returns

<NR3>

Examples

SENSe:LTE:ACLR:STEP:MAGNITUDE 4.000000000E+6 sets the step magnitude to 4 MHz.

SENSe:LTE:ANALysis:AVErage:OVER:OFFSlot:STATe

Determines whether to enable or disable Auto check box of Average over offslot in the Prefs tab of LTE Power vs Time display control panel.

Conditions

Measurement view: LTE Power vs Time

Group

Sense commands

Syntax

```
SENSe:LTE:ANALysis:AVErage:OVER:OFFSlot:STATe {0 | OFF | 1 | ON}
SENSe:LTE:ANALysis:AVErage:OVER:OFFSlot:STATe?
```

Arguments

0 or OFF disables the Average over offslot check box

1 or ON enables the Average over offslot check box

Returns

<Boolean> 0, 1

Examples

SENSe:LTE:ANALYSIS:AVERAGE:OVER:OFFSLOT:STATE OFF disables the Average over offslot check box.

SENSe]:LTE:ANALysis:EQUAlization:STATe

Sets or queries the Enable Equalization state in the Analysis Params tab of the LTE Settings control panel.

Conditions

Measurement view: LTE Constellation, LTE Power vs Time

Group

Sense commands

Syntax

```
SENSe]:LTE:ANALysis:EQUAlization:STATe { 0 | 1 }
SENSe]:LTE:ANALysis:EQUAlization:STATe?
```

Arguments

`0` disables the Enable Equalization.

`1` enables the Enable Equalization.

Returns

0 means that Enable Equalization is disabled. 1 means that Enable Equalization is enabled.

Examples

`SENSe:LTE:ANALYSIS:EQUALIZATION:STATE?` might return `0`, indicating that Enable Equalization is disabled.

SENSe:LTE:ANALysis:LENGth

Sets or queries the Analysis length value in the Analysis Time tab of the LTE Power vs Time display and LTE Constellation display control panels.

Conditions

Measurement view: LTE Power vs Time, LTE Constellation

Group

Sense commands

Syntax

```
SENSe:LTE:ANALysis:LENGth <Value>
SENSe:LTE:ANALysis:LENGth?
```

Arguments

`<Value>=<NR2>`

Returns

`<NR2>`

Examples

`SENSe:LTE:ANALYSIS:LENGTH?` might return `34.2`, indicating the analysis length is 34.2.

SENSe:LTE:ANALysis:LENGth:ACTual? (Query Only)

Queries the actual analysis length for the LTE Constellation and LTE Power vs Time displays. A query returns a value in either seconds or symbols.

Conditions

Measurement view: LTE Constellation, LTE Power vs Time

Group

Sense commands

Syntax

```
SENSe:LTE:ANALysis:LENGth:ACTual?
```

Returns

<NRf>

Examples

SENSe:LTE:ANALYSIS:LENGTH:ACTUAL? might return 43, indicating that the actual analysis length is 43 symbols.

SENSe:LTE:ANALysis:LENGth:AUTO

Determines whether to enable or disable the Auto check box for Analysis Length in the Analysis Time tab of the LTE Power vs Time display and LTE Constellation display control panels.

Conditions

Measurement view: LTE Power vs Time, LTE Constellation

Group

Sense commands

Syntax

```
SENSe:LTE:ANALysis:LENGth:AUTO {0 | OFF |1 | ON}
SENSe:LTE:ANALysis:LENGth:AUTO?
```

Arguments

0 or OFF disables the Auto checkbox of analysis length

1 or ON enables the Auto checkbox of analysis length

Returns

<Boolean> 0, 1

Examples

SENSe:LTE:ANALYSIS:LENGTH:AUTO ON enables the Analysis Length Auto check box.

[SENSe]:LTE:ANALysis:OFFSet

Sets or queries the LTE analysis offset for the LTE Constellation or LTE Power vs Time measurements. The query command returns the result in symbols or seconds, depending on the unit choice selected in the Analysis Time tab of the LTE Settings control panel.

Conditions

Measurement view: LTE Constellation, LTE Power vs Time

Group

Sense commands

Syntax

```
[SENSe]:LTE:ANALysis:OFFSet <value>
```

Arguments

<value> specifies the analysis offset in the Analysis Time tab of the LTE Settings control panel.

Returns

<NRf> Units are determined from the units choice selected in the Analysis Time tab of the LTE Settings control panel.

Examples

`SENSe:LTE:ANALYSIS:OFFSET -50` sets the analysis offset to −50 symbols.

SENSe:LTE:ANALysis:OFFSet:AUTO

Sets or queries the Auto checkbox state for analysis offset. This setting determines if the analysis offset for the LTE Constellation and LTE Power vs Time displays is set automatically or manually.

Conditions

Measurement view: LTE Constellation, LTE Power vs Time

Group

Sense commands

Syntax

```
SENSe:LTE:ANALysis:OFFSet:AUTO { 0 | 1 }
```

Arguments

`0` sets the analysis offset to manual (box is unchecked)

`1` sets the analysis offset to automatic (checkbox is checked)

Returns

0 means the analysis offset is set to manual (box is unchecked)

1 means the analysis offset is set to automatic (checkbox is checked)

Examples

`SENSe:LTE:ANALYSIS:OFFSET:AUTO 1` sets the analysis offset to automatic.

SENSe:LTE:ANALysis:TIMe:UNIT

Sets or queries the unit preference under the Analysis tab of the LTE Settings control panel for the LTE Constellation and LTE Power vs Time displays.

Conditions

Measurement view: LTE Constellation, LTE Power vs Time

Group

Sense commands

Syntax

```
SENSe:LTE:ANALysis:TIMe:UNIT { SEC | SYMB }
SENSe:LTE:ANALysis:TIMe:UNIT?
```

Arguments

`SEC` sets the units to seconds.

`SYMB` sets the units to symbols.

Returns

SEC means the units are set to seconds.

SYMB means the units are set to symbols.

Examples

`SENSe:LTE:ANALYSIS:TIME:UNIT SEC` sets the units to seconds.

SENSe:LTE:ANTEnna:PORTs:CHOSen

Sets or queries the chosen port in the Modulation Params tab of the LTE Power vs Time display and LTE Constellation display control panels.

Conditions

Measurement view: LTE Power vs Time, LTE Constellation

Group

Sense commands

Syntax

```
SENSe:LTE:ANTEnna:PORTs:CHOSen {ZERO|ONE|TWO|THREE}
SENSe:LTE:ANTEnna:PORTs:CHOSen?
```

Arguments

`{ZERO|ONE|TWO|THREE}`

Returns

`<String> (ZERO to THREE)`

Examples

`SENSe:LTE:ANTENNA:PORTS:CHOSEN ONE` sets the chosen port to port 1.

SENSe:LTE:ANTEnna:PORTs:TOTal

Sets or queries the total number of Antenna ports in the Modulation Params tab of the LTE Power vs Time display and LTE Constellation display control panels.

Conditions

Measurement view: LTE Power vs Time, LTE Constellation

Group

Sense commands

Syntax

```
SENSe:LTE:ANTEnna:PORTs:TOTal {ONE|TWO|FOUR}
SENSe:LTE:ANTEnna:PORTs:TOTal?
```

Arguments

`{ONE|TWO|FOUR}`

Returns

`<String> ONE , TWO or FOUR`

Examples

`SENSe:LTE:ANTENNA:PORTS:TOTAL FOUR` sets the total number of Antenna ports to 4.

SENSe:LTE:AUTO:DETect

Determines whether to enable or disable the Auto check box in the Modulation Params tab of the LTE Power vs Time display and LTE Constellation display control panel to automatically detect.

Conditions

Measurement view: LTE Power vs Time, LTE Constellation

Group

Sense commands

Syntax

```
SENSe:LTE:AUTO:DETect {0 | OFF | 1 | ON}
SENSe:LTE:AUTO:DETect?
```

Arguments

`ON` or `1` to enables the Auto check box

`OFF` or `0` to disables the Auto check box

Returns

`<Boolean> 0, 1`

Examples

`SENSe:LTE:AUTO:DETECT 1` enables the Auto detect check box.

SENSe:LTE:CHANnel:(BANDwidth|BWIDth)

Sets or queries Channel Bandwidth value in the Modulation Params tab of the LTE Power vs Time display and LTE Constellation display control panels.

Conditions

Measurement view: LTE Power vs Time, LTE Constellation

Group

Sense commands

Syntax

```
SENSe:LTE:CHANnel:(BANDwidth|BWIDth) {BW1PT4M|BW3M| BW5M| BW10M| BW15M| BW20M}
SENSe:LTE:CHANnel:(BANDwidth|BWIDth)?
```

Arguments

`BW1PT4M` sets the value to 1.4MHz

`BW3M` sets the value to 3 MHz

`BW5M` sets the value to 5 MHz

`BW10M` sets the value to 10 MHz

`BW15M` sets the value to 15 MHz

`BW20M` sets the value to 20 MHz

Returns

<String> See Arguments

Examples

SENSe:LTE:CHANNEL:(BANDWIDTH|BWIDTH) BW15M sets the Channel Bandwidth to 15 MHz.

SENSe:LTE:CHANnel:BANDwidth

Specifies or queries the channel bandwidth to use for the LTE Constellation or LTE Power vs Time measurements.

Conditions

Measurement view: LTE Constellation, LTE Power vs Time

Group

Sense commands

Syntax

```
SENSe:LTE:CHANnel:BANDwidth { BW5M | BW10M | BW15M | BW20M }
```

Arguments

BW5M, BW10M, BW15M, BW20M each specifies a channel bandwidth to use for the measurement.

Returns

Query returns one of the argument values, indicating the selected channel bandwidth.

Examples

SENSe:LTE:CHANNEL:BANDWIDTH BW5M sets the channel bandwidth to use to 5 MHz.

SENSe:LTE:CHSPectrum:AVERage

Sets or queries how to average waveform in the LTE Channel Spectrum display.

Conditions

Measurement view: LTE Channel Spectrum

Group

Sense commands

Syntax

```
SENSe:LTE:CHSPectrum:AVERage { OFF | TIME | FREQ }
SENSe:LTE:CHSPectrum:AVERage?
```

Arguments

OFF disables averaging.

TIME averages the waveform using time samples.

FREQuency averages the waveform using frequency samples.

Returns

OFF means averaging is disabled.

TIME means averaging of the waveform is performed using time samples.

FREQuency means averaging of the waveform is performed using frequency samples.

Examples

SENSe:LTE:CHSPECTRUM:AVERAGE TIME sets the analyzer to average the waveform using time samples.

SENSe:LTE:CHSPectrum:AVERage:COUNt

Sets or queries the number of traces for averaging in the LTE Channel Spectrum display.

Conditions

Measurement view: LTE Channel Spectrum

Group

Sense commands

Syntax

```
SENSe:LTE:CHSPectrum:AVERage:COUNt <number>
SENSe:LTE:CHSPectrum:AVERage:COUNt?
```

Arguments

<number>::<NR1> specifies the average count

Returns

<NR1>

Examples

SENSe:LTE:CHSPECTRUM:AVERAGE:COUNT? might return 10, indicating that 10 is the number of traces for averaging.

SENSe:LTE:CHSPectrum:BANDwidth:RESolution

Sets or queries the resolution bandwidth (RBW) in the LTE Channel Spectrum display.

Conditions

Measurement view: LTE Channel Spectrum

Group

Sense commands

Syntax

```
SENSe:LTE:CHSPectrum:BANDwidth:RESolution
```

Arguments

<value>::<NRf> specifies the resolution bandwidth

Returns

<NR3>

Examples

SENSe:LTE:CHSPECTRUM:BANDWIDTH:RESOLUTION? might return 30.0000000E+3, indicating the RBW is 30 kHz.

SENSe:LTE:CHSPectrum:BANDwidth:RESOlution:AUTO

Determines whether to set the resolution bandwidth (RBW) in the LTE Channel Spectrum display automatically or manually.

Conditions

Measurement view: LTE Channel Spectrum

Group

Sense commands

Syntax

```
SENSe:LTE:CHSPectrum:BANDwidth:RESOlution:AUTO { OFF | ON| 0 | 1 }
SENSe:LTE:CHSPectrum:BANDwidth:RESOlution:AUTO?
```

Arguments

OFF or 0 sets the RBW to manual.

ON or 1 sets the RBW to automatic.

Returns

OFF or 0 means that the RBW is set to manual.

ON or 1 means that the RBW is set to automatic.

Examples

SENSe:LTE:CHSPECTRUM:BANDWIDTH:RESOLUTION:AUTO 1 sets the RBW to automatic and is decided by the analysis module.

SENSe:LTE:CHSPectrum:BANDwidth:VIDeo

Sets or queries the video bandwidth (VBW) in the LTE Channel Spectrum display.

Conditions

Measurement view: LTE Channel Spectrum

Group

Sense commands

Syntax

```
SENSe:LTE:CHSPectrum:BANDwidth:VIDeo <value>
```

Arguments

<value>::<NRf> specifies the video bandwidth

Returns

<NR3>

Examples

SENSe:LTE:CHSPECTRUM:BANDWIDTH:VIDEO 3.0000000E+3 sets the VBW to 3 kHz.

SENSe:LTE:CHSPectrum:BANDwidth:VIDeo:STATe

Sets or queries enabling or disabling the video bandwidth (VBW) in the LTE Channel Spectrum display.

Conditions

Measurement view: LTE Channel Spectrum

Group

Sense commands

Syntax

```
SENSe:LTE:CHSPectrum:BANDwidth:VIDeo:STATe { 0 | 1 }
SENSe:LTE:CHSPectrum:BANDwidth:VIDeo:STATe?
```

Arguments

`0` disables the VBW.

`1` enables the VBW.

Returns

0 means that the VBW is disabled.

1 means that the VBW is enabled.

Examples

`SENSe:LTE:CHSPECTRUM:BANDWIDTH:VIDEO:STATE 1` sets the VBW to enabled.

SENSe:LTE:CHSPectrum:CHANnel:BANDwidth

Specifies or queries a nominal channel bandwidth to use for the LTE Channel Spectrum measurements.

Conditions

Measurement view: LTE Channel Spectrum

Group

Sense commands

Syntax

```
SENSe:LTE:CHSPectrum:CHANnel:BANDwidth { BW5M | BW10M | BW20M | BW1PT4M | BW3M | BW15M }
```

Arguments

`BW5M`, `BW10M`, `BW20M`, `BW1PT4M`, `BW3M`, `BW15M` each specifies a nominal channel bandwidth to use for the measurement.

Returns

Query returns one of the argument values, indicating the selected nominal channel bandwidth.

Examples

`SENSe:LTE:CHSPECTRUM:CHANNEL:BANDWIDTH BW20M` sets the nominal channel bandwidth to use to 20 MHz.

SENSe:LTE:CHSPectrum:CLEar:RESults (No Query Form)

Clears the results in the LTE Channel Spectrum measurement.

Conditions

Measurement view: LTE Channel Spectrum

Group

Sense commands

Syntax

```
SENSe:LTE:CHSPectrum:CLEar:RESults
```

Arguments

None

Examples

`SENSe:LTE:CHSPECTRUM:CLEAR:RESULTS` clears the results of the LTE Channel Spectrum measurement.

SENSe:LTE:CHSPectrum:DETection

Sets or queries the trace detection settings for the LTE Channel Spectrum display.

Conditions

Measurement view: LTE Channel Spectrum

Group

Sense commands

Syntax

```
SENSe:LTE:CHSPectrum:DETection { POS | AVER }
SENSe:LTE:CHSPectrum:DETection?
```

Arguments

`AVERage`: sets the detection setting to Average (VRMS) POS: sets the detection setting to the positive peak

Returns

`AVERage` means the detection setting is Average (VRMS) POS means the detection setting is positive peak

Examples

`SENSe:LTE:CHSPECTRUM:DETECTION POS` sets the trace detection settings to +Peak under the Prefs tab.

SENSe:LTE:CHSPectrum:FREQuency

Sets or queries the measurement center frequency in the LTE Channel Spectrum display.

Conditions

Measurement view: LTE Channel Spectrum

Group

Sense commands

Syntax

```
SENSe:LTE:CHSPectrum:FREQuency <value>
```

SENSe:LTE:CHSPectrum:FREQuency?

Arguments

<value>::<NRf> specifies the center frequency

Returns

<NR3>

Examples

SENSe:LTE:CHSPECTRUM:FREQUENCY 1.000000000E+9 sets the center frequency to 1.0 GHz.

SENSe:LTE:CHSPectrum:FREQuency:SPAN

Sets or queries the frequency span in the LTE Channel Spectrum measurement.

Conditions

Measurement view: LTE Channel Spectrum

Group

Sense commands

Syntax

SENSe:LTE:CHSPectrum:FREQuency:SPAN <value>

SENSe:LTE:CHSPectrum:FREQuency:SPAN?

Arguments

<value>::<NRf> specifies the frequency span

Returns

<NRf>

Examples

SENSe:LTE:CHSPECTRUM:FREQUENCY:SPAN sets the frequency span in the LTE Channel Spectrum display.

SENSe:LTE:CHSPectrum:FREQuency:STEP

Sets or queries the frequency step size in the LTE Channel Spectrum measurement. Programming a specified step size sets SENSe:LTE:CHSPectrum:FREQuency:STEP:AUTO to OFF.

Conditions

Measurement view: LTE Channel Spectrum

Group

Sense commands

Syntax

SENSe:LTE:CHSPectrum:FREQuency:STEP <value>
SENSe:LTE:CHSPectrum:FREQuency:STEP?

Arguments

<value>::<NRf> specifies the frequency step size

Returns

<NR3>

Examples

SENSe:LTE:CHSPECTRUM:FREQUENCY:STEP 200.000000000E+3 sets the frequency step size to 200 kHz.

SENSe:LTE:CHSPectrum:FREQuency:STEP:AUTO

Determines whether to set the frequency step size automatically or manually in the LTE Channel Spectrum measurement.

Conditions

Measurement view: LTE Channel Spectrum

Group

Sense commands

Syntax

```
SENSe:LTE:CHSPectrum:FREQuency:STEP:AUTO { OFF | ON | 0 | 1 }
SENSe:LTE:CHSPectrum:FREQuency:STEP:AUTO?
```

Arguments

OFF or 0 sets the frequency step size is set to manual.

ON or 1 sets the frequency step size is set to automatic.

Returns

OFF or 0 means that the frequency step size is set to manual. ON or 1 means that the frequency step size is set to automatic.

Examples

SENSe:LTE:CHSPECTRUM:FREQUENCY:STEP:AUTO ON specifies that the frequency step size is set automatically.

SENSe:LTE:CHSPectrum:POINts:COUNt

Sets or queries the number of points used for the trace in the LTE Channel Spectrum measurement.

Conditions

Measurement view: LTE Channel Spectrum

Group

Sense commands

Syntax

```
SENSe:LTE:CHSPectrum:POINts:COUNt <value>
SENSe:LTE:CHSPectrum:POINts:COUNt?
```

Arguments

<value>::<NRf> specifies the number of points used for the trace

Returns

P<NR1>

Examples

SENSe:LTE:CHSPECTRUM:POINTS:COUNT? might return P8001, indicating a maximum of 8001 trace points can be used.

SENSe:LTE:CYClic:PREFix

Sets or queries the cyclic prefix type in the Modulation Params tab of the LTE Power vs Time display and LTE Constellation display control panels.

Conditions

Measurement view: LTE Power vs Time, LTE Constellation

Group

Sense commands

Syntax

```
SENSe:LTE:CYClic:PREFix {NORM|EXT}
SENSe:LTE:CYClic:PREFix?
```

Arguments

NORM sets the cyclic prefix to Normal.

EXT sets the cyclic prefix to Extended.

Returns

<String> NORM or EXT

Examples

SENSe:LTE:CYCLIC:PREFIX NORM sets the cyclic prefix to Normal.

SENSe:LTE:FRAMe:STRUcture

Sets or queries the frame structure in the Modulation Params tab of the LTE settings control panel.

Conditions

Measurement view: LTE Constellation and LTE Power vs Time

Group

Sense commands

Syntax

```
SENSe:LTE:FRAMe:STRUcture { FDD | TDD }
SENSe:LTE:FRAMe:STRUcture?
```

Arguments

FDD

TDD

Returns

FDD TDD

Examples

SENSe:LTE:FRAME:STRUCTURE FDD sets the frame structure to FDD.

SENSe:LTE:LINK:CONFiguration

Sets or queries Uplink-Downlink value in TDD configuration in the Modulation Params tab of the LTE Power vs Time display and LTE Constellation display control panels.

This happens only when Downlink TDD is Selected as Frame Structure

Conditions

Measurement view: LTE Power vs Time, LTE Constellation

Group

Sense commands

Syntax

```
SENSe:LTE:LINK:CONFiguration {ZERO |ONE |TWO |THREE |FOUR|FIVE |SIX}
SENSe:LTE:LINK:CONFiguration?
```

Arguments

{ZERO |ONE |TWO |THREE |FOUR |FIVE |SIX}

Returns

<String> (Zero to Six)

Examples

SENSe:LTE:LINK:CONFIGURATION ONE sets the Uplink-Downlink value as 1 in TDD configuration.

SENSe:LTE:SUBFrame:CONFiguration

Sets or queries special subframe value in TDD configuration in the Modulation Params tab of the LTE Power vs Time display and LTE Constellation display control panels.

This happens only when Downlink TDD is selected as the Frame Structure

Conditions

Measurement view: LTE Power vs Time, LTE Constellation

Group

Sense commands

Syntax

```
SENSe:LTE:SUBFrame:CONFiguration {ZERO |ONE |TWO |THREE|FOUR |FIVE |SIX |SEVEN |EIGHT |NINE}
SENSe:LTE:SUBFrame:CONFiguration?
```

Arguments

{ZERO |ONE |TWO |THREE |FOUR |FIVE |SIX |SEVEN |EIGHT |NINE}

Returns

<String> (Zero to Nine)

Examples

SENSe:LTE:SUBFRAME:CONFIGURATION SIX sets the special subframe value to 6 in TDD configuration.

[SENSe]:MCPower:AVERage

Sets or queries the average method in the MCPR measurement.

Conditions

Measurement views: MCPR

Group

Sense commands

Syntax

```
[SENSe]:MCPower:AVERage { OFF | TIMe | FREQuency }
[SENSe]:MCPower:AVERage?
```

Arguments

OFF disables averaging.

TIMe performs averaging for time samples.

FREQuency performs averaging for frequency samples.

Examples

SENSe:MCPOWER:AVERAGE TIMe performs averaging for time samples.

[SENSe]:MCPower:AVERage:COUNt

Sets or queries the average count in the MCPR measurement.

Conditions

Measurement views: MCPR

Group

Sense commands

Syntax

```
[SENSe]:MCPower:AVERage:COUNt <value>
[SENSe]:MCPower:AVERage:COUNt?
```

Arguments

<value>::<NR1> specifies the average count. Range: 2 to 10000.

Examples

SENSe:MCPOWER:AVERAGE:COUNT 256 sets the average count to 256.

[SENSe]:MCPower:{BANDwidth|BWIDth}[:RESolution]

Sets or queries the resolution bandwidth (RBW). Programming a specified RBW sets `[SENSe]:MCPower{BANDwidth|BWIDth}[:RESolution]:AUTO OFF`.

Conditions

Measurement views: MCPR

Group

Sense commands

Syntax

```
[SENSe]:MCPower:{BANDwidth|BWIDth}[:RESolution] <value>
[SENSe]:MCPower:{BANDwidth|BWIDth}[:RESolution]?
```

Related Commands

[SENSe]:MCPower:{BANDwidth|BWIDth}[:RESolution]:AUTO (on page 2180)

Arguments

`<value>::<NRf>` specifies the RBW. Range: 100 Hz to 5 MHz.

Examples

`SENSe:MCPOWER:BANDWIDTH:RESOLUTION 200kHz` sets the RBW to 200 kHz.

[SENSe]:MCPower:{BANDwidth|BWIDth}[:RESolution]:ACTual? (Query Only)

Queries the actual resolution bandwidth (RBW) in the MCPR measurement.

Conditions

Measurement views: MCPR

Group

Sense commands

Syntax

```
[SENSe]:MCPower:{BANDwidth|BWIDth}[:RESolution]:ACTual?
```

Arguments

None

Returns

`<NRf>` The actual RBW in Hz.

Examples

`SENSe:MCPOWER:BANDWIDTH:RESOLUTION:ACTUAL?` might return `299.624E+3`, indicating that the actual RBW is 299.624 kHz.

[SENSe]:MCPower:{BANDwidth|BWIDth}[:RESolution]:AUTO

Determines whether to set the resolution bandwidth (RBW) automatically or manually.

Conditions

Measurement views: MCPR

Group

Sense commands

Syntax

```
[SENSe]:MCPower:{BANDwidth|BWIDth}[:RESolution]:AUTO { OFF | ON | 0 | 1 }
[SENSe]:MCPower:{BANDwidth|BWIDth}[:RESolution]:AUTO?
```

Arguments

`OFF` or `0` specifies that the resolution bandwidth is set manually using the `[SENSe]:MCPower:{BANDwidth|BWIDth}[:RESolution]` command.

`ON` or `1` specifies that the resolution bandwidth is set automatically.

Examples

`SENSe:MCPOWER:BANDWIDTH:AUTO ON` sets the resolution bandwidth automatically.

[SENSe]:MCPower:{BANDwidth|BWIDth}:VIDeo

Sets or queries the video bandwidth (VBW). Programming a specified VBW sets `[SENSe]:MCPower:{BANDwidth|BWIDth}:VIDeo:STATe OFF`.

Conditions

Measurement views: MCPR

Group

Sense commands

Syntax

```
[SENSe]:MCPower:{BANDwidth|BWIDth}:VIDeo <value>
[SENSe]:MCPower:{BANDwidth|BWIDth}:VIDeo?
```

Related Commands

[SENSe]:MCPower:{BANDwidth|BWIDth}:VIDeo:STATe (on page 2180)

Arguments

`<value>::<NRf>` specifies the VBW.

Range: Current RBW/104 (1 Hz minimum) to Current RBW.

Examples

`SENSe:MCPOWER:BANDWIDTH:VIDEO 200kHz` sets the VBW to 200 kHz.

[SENSe]:MCPower:{BANDwidth|BWIDth}:VIDeo:STATe

Determines whether to enable the video bandwidth (VBW) in the MCPR measurement.

Conditions

Measurement views: MCPR

Group

Sense commands

Syntax

```
[SENSe]:MCPower:{BANDwidth|BWIDth}:VIDeo:STATe { OFF | ON| 0 | 1 }
[SENSe]:MCPower:{BANDwidth|BWIDth}:VIDeo:STATe?
```

Arguments

`OFF` or `0` disables the VBW.

`ON` or `1` enables the VBW.

Examples

`SENSe:MCPOWER:BANDWIDTH:VIDEO:STATE ON` enables the VBW.

[SENSe]:MCPower:CHANnel:ADJacent:ADD (No Query Form)

Adds a pair of upper and lower adjacent channels in the MCPR measurement.

Conditions

Measurement views: MCPR

Group

Sense commands

Syntax

```
[SENSe]:MCPower:CHANnel:ADJacent:ADD <offset>,<bandwidth>
```

Arguments

`<offset>::<NRf>` specifies the offset from the center frequency for the adjacent channel.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

`<bandwidth>::<NRf>` specifies the bandwidth of the adjacent channel.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples

`SENSe:MCPOWER:CHANNEL:ADJACENT:ADD 200kHz, 80kHz` adds a pair of upper and lower adjacent channels with the offset of ±200 kHz and the bandwidth of 80 kHz.

[SENSe]:MCPower:CHANnel:ADJacent:DELete (No Query Form)

Deletes a selected adjacent channel in the MCPR measurement.

Conditions

Measurement views: MCPR

Group

Sense commands

Syntax

```
[SENSe]:MCPower:CHANnel:ADJacent:DELete <channel>
```

Arguments

`<channel>::<string>` specifies the channel to be deleted.

Specify the channel with "A<n>" for the adjacent channel where <n> represents the channel number (<n> = 1, 2, 3,...). See the example below.

Examples

`SENSe:MCPower:CHANnel:ADJacent:DELete "A2"` deletes A2 (the adjacent channel 2).

[SENSe]:MCPower:CHANnel:ADJacent:LIMIts<x>

This command returns the selected adjacent upper and lower limit values. Selection of adjacent channel.

Conditions

Measurement view: P25 MCPR

The parameter <x>=1 to 100.

Group

Sense commands

Syntax

```
[SENSe]:MCPower:CHANnel:ADJacent:LIMIts<x>
[SENSe]:MCPower:CHANnel:ADJacent:LIMIts<x>?
```

Arguments

`<x>`=1 to 100

Returns

Upper and lower limit values for selected adjacent channels.

Examples

`SENSe:MCPower:CHANnel:ADJacent:LIMIts2?` returns upper and lower limit values of A2 (Adjacent channel 2).

[SENSe]:MCPower:CHANnel:ADJacent:LIMIts<x>:ENABle? (Query Only)

This query returns whether the adjacent channel is selected or not.

Conditions

Measurement view: P25 MCPR

Group

Sense commands

Syntax

```
[SENSe]:MCPower:CHANnel:ADJacent:LIMIts<x>:ENABle?
```

Returns

`0` means the adjacent channel is disabled.

`1` means the adjacent channel is enabled.

Examples

If `SENSe:MCPower:CHANnel:ADJacent:LIMIts1:ENABle?` returns `1`, A1 (adjacent channel 1) is enabled.

[SENSe]:MCPower:CHANnel:FILTer

Sets or queries the adjacent channel filter in the MCPR measurement.

Conditions

Measurement views: MCPR

Group

Sense commands

Syntax

```
[SENSe]:MCPower:CHANnel:FILTer { RRCosine | NONE }
[SENSe]:MCPower:CHANnel:FILTer?
```

Arguments

`RRCosine` selects the Root-Raised-Cosine filter.

`NONE` uses no filter.

Examples

`SENSe:MCPOWER:CHANNEL:FILTER RRCosine` selects Root-Raised-Cosine for the adjacent channel filter.

[SENSe]:MCPower:CHANnel:MAIN:{BANDwidth|BWIDth}

Sets or queries the frequency bandwidth of the main channels (all share the same value) in the MCPR measurement.

Conditions

Measurement views: MCPR

Group

Sense commands

Syntax

```
[SENSe]:MCPower:CHANnel:MAIN:{BANDwidth|BWIDth} <value>
[SENSe]:MCPower:CHANnel:MAIN:{BANDwidth|BWIDth}?
```

Arguments

<value>::<NRf> specifies the main channel bandwidth.

Range: 1 Hz to full span.

Examples

SENSe:MCPOWER:CHANNEL:MAIN:BANDWIDTH 4.5MHz sets the main channel bandwidth to 4.5 MHz.

[SENSe]:MCPower:CHANnel:MAIN:COUNt

Sets or queries the number of main channels in the MCPR measurement. You can use this command to add and remove main channels.

Conditions

Measurement views: MCPR

Group

Sense commands

Syntax

```
[SENSe]:MCPower:CHANnel:MAIN:COUNt <value>
[SENSe]:MCPower:CHANnel:MAIN:COUNt?
```

Arguments

<value>::<NRf> specifies the number of main channels. Range: 1 to 99.

Examples

SENSe:MCPOWER:CHANNEL:MAIN:COUNT 3 sets the the number of main channels to 3.

[SENSe]:MCPower:CHANnel:MAIN:INACtive

Makes a specified main channel inactive. You can set it on or off. The query returns all inactive main channels.

Conditions

Measurement views: MCPR

Group

Sense commands

Syntax

```
[SENSe]:MCPower:CHANnel:MAIN:INACtive <channel>,<boolean>
[SENSe]:MCPower:CHANnel:MAIN:INACtive?
```

Arguments

<channel>::<string> specifies the channel to be inactive.

Specify the channel with "M<n>" for the main channel where <n> represents the channel number (<n> = 1, 2, 3,...). See the example below.

<boolean>::={ OFF | ON | 0 | 1 } specifies that the specified channel is inactive (On) or not (Off).

Examples

SENSe:MCPOWER:CHANNEL:MAIN:INACTIVE "M2",ON makes the main channel 2 inactive.

[SENSe]:MCPower:CHANnel:MAIN:SPACing

Sets or queries frequency difference between centers of each main channel in the MCPR measurement.

Conditions

Measurement views: MCPR

Group

Sense commands

Syntax

```
[SENSe]:MCPower:CHANnel:MAIN:SPACing <value>
[SENSe]:MCPower:CHANnel:MAIN:SPACing?
```

Arguments

<value>::<NRf> specifies the spacing between two adjacent main channels.

Range: 1 Hz to 1 GHz.

Examples

SENSe:MCPOWER:CHANNEL:MAIN:SPACING 5MHz sets the main channel spacing to 5 MHz.

[SENSe]:MCPower:CHIPrate

Sets or queries the chip rate in the MCPR measurement. This command is valid when [SENSe]:MCPower:CHANnel:FILTer is set to RRCosine (Root-Raised-Cosine).

Conditions

Measurement views: MCPR

Group

Sense commands

Syntax

```
[SENSe]:MCPower:CHIPrate <value>
[SENSe]:MCPower:CHIPrate?
```

Arguments

<value>::<NRf> specifies the chip rate. Range: 100 Hz to 105 MHz.

Examples

SENSe:MCPOWER:CHIPRATE 1kHz sets the chip rate to 1 kHz.

[SENSe]:MCPower:CLEar:RESults (No Query Form)

Restarts the average trace.

Conditions

Measurement views: MCPR

Group

Sense commands

Syntax

```
[SENSe]:MCPower:CLEar:RESults
```

Arguments

None

Examples

`SENSe:MCPOWER:CLEAR:RESULTS` restarts the average trace.

[SENSe]:MCPower:FREQuency

Sets or queries the center frequency in the MCPR measurement.

Conditions

Measurement views: MCPR

Group

Sense commands

Syntax

```
[SENSe]:MCPower:FREQuency <value>
[SENSe]:MCPower:FREQuency?
```

Arguments

`<value>::<NRf>` specifies the center frequency.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B– 1 to 26.5 GHz

Examples

`SENSe:MCPOWER:FREQUENCY 2.35GHz` sets the center frequency to 2.35 GHz.

[SENSe]:MCPower:FREQuency:STEP

Sets or queries the frequency step size. Programming a specified step size sets `[SENSe]:MCPower:FREQuency:STEP:AUTO OFF`.

Conditions

Measurement views: MCPR

Group

Sense commands

Syntax

```
[SENSe]:MCPower:FREQuency:STEP <value>
[SENSe]:MCPower:FREQuency:STEP?
```

Related Commands

[SENSe]:MCPower:FREQuency:STEP:AUTO (on page 2187)

Arguments

`<value>::<NRf>` specifies the frequency step size.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106AB – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples

`SENSe:MCPOWER:FREQUENCY:STEP 50kHz` sets the frequency step size to 50 kHz.

[SENSe]:MCPower:FREQuency:STEP:AUTO

Determines whether to set the frequency step size automatically or manually in the MCPR measurement.

Conditions

Measurement views: MCPR

Group

Sense commands

Syntax

```
[SENSe]:MCPower:FREQuency:STEP:AUTO { OFF | ON | 0 | 1 }
[SENSe]:MCPower:FREQuency:STEP:AUTO?
```

Arguments

`OFF` or `0` specifies that the frequency step size is set manually using the `[SENSe]:MCPower:FREQuency:STEP` command.

`ON` or `1` specifies that the frequency step size is set automatically.

Examples

`SENSe:MCPOWER:FREQUENCY:STEP:AUTO ON` specifies that the frequency step size is set automatically.

[SENSe]:MCPower:NFLoor:STATe

Determines whether to enable correction for noise floor.

Conditions

Measurement views: MCPR

Group

Sense commands

Syntax

```
[SENSe]:MCPower:NFLoor:STATe { OFF | ON | 0 | 1 }
[SENSe]:MCPower:NFLoor:STATe?
```

Arguments

`OFF` or `0` disables correction for noise floor.

`ON` or `1` enables correction for noise floor.

Examples

`SENSe:MCPOWER:NFLOOR:STATE ON` enables correction for noise floor.

[SENSe]:MCPower:POINts:COUNt

Sets or queries the number of trace points in the MCPR display.

Conditions

Measurement views: MCPR

Group

Sense commands

Syntax

```
[SENSe]:MCPower:POINts:COUNt { P801 | P2401 | P4001 | P8001| P10801 }
[SENSe]:MCPower:POINts:COUNt?
```

Arguments

`P801 | P2401 | P4001 | P8001 | P10801` set the number of trace points.

Examples

`SENSe:MCPOWER:POINTS:COUNT P4001` sets the number of trace points in the MCPR measurement to 4001.

[SENSe]:MCPower:RCHannels? (Query Only)

Queries the power reference in the MCPR measurement.

Conditions

Measurement views: MCPR

Group

Sense commands

Syntax

```
[SENSe]:MCPower:RCHannels?
```

Arguments

None

Returns

`<power_ref>::={ Total | M<x> }`

where `<x>` = 1 to 99.

`Total` indicates that the power reference is the total power of all the active channels.

`M<x>` indicates that the power reference is the main channel with the index (<x>).

Examples

`SENSe:MCPOWER:RCHANNELS?` might return `M3`, indicating that the power reference is the main channel.

[SENSe]:MCPower:RCHannels:MAIN<x> (No Query Form)

Sets the power reference to the main channel with the index (<x>) in the MCPR measurement.

The parameter <x> = 1 to 99, representing the main channel 1 to 99, respectively. The main channel must be defined using the `[:SENSe]:MCPower:CHANnel:MAIN` commands.

Conditions

Measurement views: MCPR

Group

Sense commands

Syntax

```
[SENSe]:MCPower:RCHannels:MAIN<x>
```

Related Commands

[SENSe]:MCPower:CHANnel:MAIN:{BANDwidth|BWIDth} (on page 2183)

[SENSe]:MCPower:CHANnel:MAIN:COUNt (on page 2184)

[SENSe]:MCPower:CHANnel:MAIN:INACtive (on page 2184)

[SENSe]:MCPower:CHANnel:MAIN:SPACing (on page 2185)

Arguments

None

Examples

`SENSe:MCPOWER:RCHANNELS:MAIN3` selects Main 3 for the power reference channel.

[SENSe]:MCPower:RCHannels:TOTal (No Query Form)

Sets the power reference to the total power of all the active channels in the MCPR measurement.

Conditions

Measurement views: MCPR

Group

Sense commands

Syntax

```
[SENSe]:MCPower:RCHannels:TOTal
```

Arguments

None

Examples

SENSe:MCPOWER:RCHANNELS:TOTAL sets the power reference to the total power of all the active channels.

[SENSe]:MCPower:RRCRolloff

Sets or queries the filter parameter (roll-off ratio) for the Root Raised Cosine filter.

Conditions

Measurement views: MCPR

Group

Sense commands

Syntax

```
[SENSe]:MCPower:RRCRolloff <value>
[SENSe]:MCPower:RRCRolloff?
```

Related Commands

[SENSe]:MCPower:CHANnel:FILTer (on page 2183)

Arguments

<value>::<NRf> specifies the filter parameter.

Range: 0.001 to 1, 0.0001 step.

Examples

SENSe:MCPOWER:RRCROLLOFF 0.3 sets the filter parameter to 0.3.

[SENSe]:MEASurement:FREQuency[:SOURce<z>]

Sets or queries the measurement frequency.

The parameter <z> = 1 to number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Default source is Source 1.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:MEASurement:FREQuency[:SOURce<z>] <value>
[SENSe]:MEASurement:FREQuency[:SOURce<z>]?
```

Arguments

<value>::<NRf> specifies the measurement frequency.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples

`SENSe:MEASurement:FREQuency:SOURCE2? 7.5GHz` sets the measurement frequency frequency of Source 2 to 7.5 GHz.

[SENSe]:MEASurement:FREQuency[:SOURce<z>]:CENTer:LOCK

Sets or queries locking the measurement center frequency.

The parameter <z> = 1 to number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Default source is Source 1.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:MEASurement:FREQuency[:SOURce<z>]:CENTer:LOCK { OFF | ON | 0 | 1 }
[SENSe]:MEASurement:FREQuency[:SOURce<z>]:CENTer:LOCK?
```

Arguments

`OFF` or 0 disables locking the center frequency.

`ON` or 1 enables locking the center frequency.

Examples

`SENSe:MEASurement:FREQuency:SOURCE1:CENTer:LOCK ON` locks the center frequency.

[SENSe]:MERRor[:MEASview<y>]:SELEcted:SOURce

Sets or queries the Source of Magnitude error versus Time for a selected Measurement view number

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Magnitude error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Magnitude error versus Time).

Source cannot be changed when acquisition or analysis is in progress. Stop the acquisition and change the Source.

Group

Sense commands

Syntax

```
[SENSe]:MERRor[:MEASview<y>]:SELEcted:SOURce {"Source 1" | "Source 2" | "Source 3" | "Source 4"}
```

```
[SENSe]:MERRor[:MEASview<y>]:SELEcted:SOURce?
```

Arguments

The parameter <y>=1 to 8 represents the measurement index based on the number of sources of connected instruments, where the MEASview keyword is optional by default and y is 1

Example

`SENSe:MERRor:MEASview1:SELEcted:SOURce?` queries the Source for Magnitude error versus Time Measurement View 1.

[SENSe]:NOISe:AVERage:COUNt

This command returns the acquisition counts over which the noise average is computed.

Conditions

Measurement view: Noise Figure

Group

Sense commands

Syntax

```
[SENSe]:NOISe:AVERage:COUNt <integer>
[SENSe]:NOISe:AVERage:COUNt?
```

Arguments

`<integer>`

Range: 1 to 1000

Returns

`<NR3>` = a floating point value with an exponent

Examples

`SENSe:NOISE:AVERAGE:COUNT?` might return `1.0000000000E+3`, indicating the average count of the analyzer is set to 1000.

[SENSe]:NOISe:BANDwidth

This command sets or queries the resolution bandwidth (RBW) for the Noise Figure display measurement.

Conditions

Measurement view: Noise Figure

Group

Sense commands

Syntax

```
[SENSe]:NOISe:BANDwidth <NR3>
[SENSe]:NOISe:BANDwidth?
```

Arguments

`<NR3>` = a floating point value with an exponent Range: 50 Hz - 10 MHz

Returns

<NR3> = a floating point value with an exponent

Examples

SENSe:NOISE:BANDWIDTH? might return 1.0000000000E+6, indicating that the RBW is 1.00 MHz.

[SENSE]:NOISe:CONTrol:SETTling:OFF[:TIME]

This command sets or queries the analyzer off settle time for the Noise Figure display measurement.

Conditions

Measurement view: Noise Figure

Group

Sense commands

Syntax

```
[SENSE]:NOISe:CONTrol:SETTling:OFF[:TIME] <NR3>
[SENSE]:NOISe:CONTrol:SETTling:OFF[:TIME]?
```

Related Commands

[SENSE]:NOISe:CONTrol:SETTling:ON[:TIME] (on page 2193)

Arguments

<NR3> = a floating point value with an exponent Range: 0 s - 10 ks

Returns

<NR3> = a floating point value with an exponent

Examples

SENSe:NOISE:CONTROL:SETTLING:OFF:TIME? might return 200.0000000000E-3, indicating the the RSA settle time (off) is 200 ms.

[SENSE]:NOISe:CONTrol:SETTling:ON[:TIME]

This command sets or queries the analyzer on settle time for the Noise and Gain display measurements.

Conditions

Measurement views: Noise Figure, Noise Temperature, Y Factor

Group

Sense commands

Syntax

```
[SENSE]:NOISe:CONTrol:SETTling:ON[:TIME] <NR3>
[SENSE]:NOISe:CONTrol:SETTling:ON[:TIME]?
```

Related Commands

[SENSE]:NOISe:CONTrol:SETTling:OFF[:TIME] (on page 2193)

Arguments

<NR3> = a floating point value with an exponent Range: 0 s - 10 ks

Returns

<NR3> = a floating point value with an exponent

Examples

SENSe:NOISE:CONTROL:SETTLING:ON:TIME? might return 50.0000000000E-3, indicating the RSA settle time (on) is 50 ms.

[SENSe]:NOISe:CORRection:ENR:CONSTant

This command sets and queries the excess noise ratio (ENR) constant for the Noise Figure and Gain displays.

Conditions

Measurement view: Noise Figure, Y Factor, Gain, Noise Temperature, Noise Table

Group

Sense commands

Syntax

```
[SENSe]:NOISe:CORRection:ENR:CONSTant <NR2>
[SENSe]:NOISe:CORRection:ENR:CONSTant?
```

Related Commands

[SENSe]:NOISe:CORRection:ENR:MODE (on page 2194)

Arguments

<NR2> = a floating point value without an exponent

Returns

<NR2> = a floating point value without an exponent

Examples

SENSe:NOISE:CORRECTION:ENR:CONSTANT? might return 10.0000000000, indicating that the ENR constant is 10.00 dB.

[SENSe]:NOISe:CORRection:ENR:MODE

This command sets and queries the Noise Source mode for the Noise Figure and Gain displays.

Conditions

Measurement view: Noise Figure, Y Factor, Gain, Noise Temperature, Noise Table

Group

Sense commands

Syntax

```
[SENSe]:NOISe:CORRection:ENR:MODE { TABLe | CONSTant }
[SENSe]:NOISe:CORRection:ENR:MODE?
```

Related Commands

[SENSe]:NOISe:CORRection:ENR:CONSTant (on page 2194)

Arguments

TABLe sets the noise source mode to Table.

CONSTant sets the noise source mode to Constant.

Returns

TABL indicates that the noise source mode is set to Table.

CONST indicates that the noise source mode is set to Constant.

Examples

SENSe:NOISE:CORRECTION:ENR:MODE? might return TABL, indicating that the noise source mode is set to Table.

[SENSe]:NOISe:CORRection:ENR:TABLe:DATA? (Query Only)

This command queries the Noise Source model, serial number, and table values for frequency and ENR for the Noise Figure and Gain displays.

Conditions

Measurement view: Noise Figure, Y Factor, Gain, Noise Temperature, Noise Table

Group

Sense commands

Syntax

```
[SENSe]:NOISe:CORRection:ENR:TABLe:DATA?
```

Related Commands

[SENSe]:NOISe:CORRection:ENR:MODE (on page 2194)

[SENSe]:NOISe:CORRection:ENR:TABLe:ID (on page 2195)

Returns

<string> is the Noise Source model and serial number, as well as the frequency and ENR values for the noise source table.

Examples

SENSe:NOISE:CORRECTION:ENR:TABLE:DATA? might return 346B 10MHz – 18 GHz, 2037A02333, 2900000000, 15, 2900000000, 15, 2900000000, 15, 2900000000, 15, 2900000000, 15, 2900000000, 15, 2900000000, 15, 2900000000, 15, 2900000000, 15, 2900000000, 15. The first two values indicate the Noise Source model and serial numbers. All of the values that follow indicate the frequency (2900000000, meaning 2.90 GHz) and the ENR (15, meaning 15.00 dB) shown in the Noise Source table.

[SENSe]:NOISe:CORRection:ENR:TABLe:ID

This command sets or queries the Noise Source Model for the Noise Figure and Gain displays.

Conditions

Measurement view: Noise Figure, Y Factor, Gain, Noise Temperature, Noise Table

Group

Sense commands

Syntax

```
[SENSe]:NOISe:CORRection:ENR:TABLe:ID <Qstring>
[SENSe]:NOISe:CORRection:ENR:TABLe:ID?
```

Related Commands

[SENSe]:NOISe:CORRection:ENR:TABLe:DATA? (on page 2195)

Arguments

`<Qstring>` = quoted string of ASCII text

Returns

`<Qstring>` = quoted string of ASCII text

Examples

`SENSe:NOISE:CORRECTION:ENR:TABLE:ID?` might return “346B 10MHz – 18 GHz”, indicating the Noise Source model is 346B 10 MHz–18 GHz.

[SENSe]:NOISe:CORRection:ENR:TABLe:NEW (No Query Form)

This command sets the Noise Source Model for the Noise Figure and Gain displays.

Conditions

Measurement view: Noise Figure, Y Factor, Gain, Noise Temperature, Noise Table

Group

Sense commands

Syntax

```
[SENSe]:NOISe:CORRection:ENR:TABLe:NEW <Qstring>
[SENSe]:NOISe:CORRection:ENR:TABLe:NEW?
```

Related Commands

[SENSe]:NOISe:CORRection:ENR:TABLe:DATA? (on page 2195)

Arguments

`<Qstring>` = quoted string of ASCII text

Returns

`<Qstring>` = quoted string of ASCII text

Examples

`SENSe:NOISE:CORRECTION:ENR:TABLE:NEW?` might return `"346B 10 MHz – 18 GHz"`, indicating the Noise Source model is 346B 10 MHz–18 GHz.

[SENSe]:NOISe:CORRection:ENR:TABLe:SERial

This command sets or queries the Noise Source serial number for the Noise Figure and Gain displays.

Conditions

Measurement view: Noise Figure, Y Factor, Gain, Noise Temperature, Noise Table

Group

Sense commands

Syntax

```
[SENSe]:NOISe:CORRection:ENR:TABLe:SERial <Qstring>
[SENSe]:NOISe:CORRection:ENR:TABLe:SERial?
```

Related Commands

[SENSe]:NOISe:CORRection:ENR:TABLe:DATA? (on page 2195)

Arguments

`<Qstring>` = quoted string of ASCII text

Returns

`<Qstring>` = quoted string of ASCII text

Examples

`SENSe:NOISE:CORRECTION:ENR:TABLE:SERIAL?` might return “2037A02334”, indicating the Noise Source serial number is 2037A02334.

[SENSe]:NOISe:CORRection:TCOLd:ADUT:UREFerence

Sets or queries the analyzer to use the reference temperature (K) for the loss at the DUT output in the noise displays.

Conditions

Measurement views: Noise Figure, Gain, Noise Temperature, Y Factor, Noise Table

Group

Sense commands

Syntax

```
[SENSe]:NOISe:CORRection:TCOLd:ADUT:UREFerence { OFF | ON| 1 | 0 }
[SENSe]:NOISe:CORRection:TCOLd:ADUT:UREFerence?
```

Arguments

`OFF` or `0` disables the Use Ref temperature feature for the Loss at DUT Output.

`ON` or `1` enables the Use Ref temperature feature for the Loss at DUT Output.

Returns

0 means that the Use Ref temperature feature for the Loss at DUT Output is disabled.

1 means that the Use Ref temperature feature for the Loss at DUT Output is enabled.

Examples

SENSe:NOIS:CORR:TCOL:ADUT:UREF? might return 1, indicating that the temperature for the loss at the DUT output is set to Use Reference temperature.

[SENSe]:NOISe:CORRection:TCOLd:ADUT[:VALue]

Sets or queries the temperature value (K) for the loss at the DUT output in the noise displays.

Conditions

Measurement views: Noise Figure, Gain, Noise Temperature, Y Factor, Noise Table

Group

Sense commands

Syntax

```
[SENSe]:NOISe:CORRection:TCOLd:ADUT[:VALue] <NR3>
[SENSe]:NOISe:CORRection:TCOLd:ADUT[:VALue]?
```

Arguments

<NR3> = floating point value with an exponent.

Range: 0 K to 500 K

Returns

<NR2> = floating point value without an exponent, which is the temperature in Kelvin.

Examples

SENSe:NOIS:CORR:TCOL:ADUT:VAL? might return 100.0000000000, indicating that the temperature for the loss at the DUT output is 100 K.

[SENSe]:NOISe:CORRection:TCOLd:BDUT:UREFerence

Sets or queries the analyzer to use the reference temperature (K) for the loss at the DUT input in the noise displays.

Conditions

Measurement views: Noise Figure, Gain, Noise Temperature, Y Factor, and Noise Table

Group

Sense commands

Syntax

```
[SENSe]:NOISe:CORRection:TCOLd:BDUT:UREFerence { OFF | ON| 1 | 0 }
[SENSe]:NOISe:CORRection:TCOLd:BDUT:UREFerence?
```

Arguments

OFF or 0 disables the Use Ref temperature feature for the Loss at DUT Input.

ON or 1 enables the Use Ref temperature feature for the Loss at DUT Input.

Returns

0 means that the Use Ref temperature feature for the Loss at DUT Input is disabled.

1 means that the Use Ref temperature feature for the Loss at DUT Input is enabled.

Examples

SENSe:NOIS:CORR:TCOL:BDUT:UREF? might return 1, indicating that the temperature for the loss at the DUT input is set to Use Reference temperature.

[SENSe]:NOISe:CORRection:TCOLd:BDUT[:VALue]

Sets or queries the temperature value (K) for the loss at the DUT input in the noise displays.

Conditions

Measurement views: Noise Figure, Gain, Noise Temperature, Y Factor, Noise Table

Group

Sense commands

Syntax

```
[SENSe]:NOISe:CORRection:TCOLd:BDUT[:VALue] <NR3>
[SENSe]:NOISe:CORRection:TCOLd:BDUT[:VALue]?
```

Arguments

<NR3> = floating point value with an exponent.

Range: 0 K to 500 K

Returns

<NR2> = floating point value without an exponent, which is the temperature in Kelvin.

Examples

SENSe:NOIS:CORR:TCOL:BDUT:VAL 290 sets the temperature for the loss at the DUT input to 290 K.

[SENSe]:NOISe:CORRection:TCOLd[:REFerence]

Sets or queries the reference temperature value (K) in the noise display.

Conditions

Measurement views: Noise Figure, Gain, Noise Temperature, Y Factor, Noise Table

Group

Sense commands

Syntax

```
[SENSe]:NOISe:CORRection:TCOLd[:REFerence] <NR3>
[SENSe]:NOISe:CORRection:TCOLd[:REFerence]?
```

Arguments

<NR3> = floating point value with an exponent.

Range: 0 K to 500 K

Returns

<NR2> = floating point value without an exponent, which is the reference temperature in Kelvin.

Examples

SENSe:NOIS:CORR:TCOL:REF 200 sets the reference temperature to 200 K.

[SENSe]:NOISe:FIGure:POINTs:COUNt (No Query Form)

Sets the number of sample points for the Noise Figure display.

Conditions

Measurement views: Noise Figure

Group

Sense commands

Syntax

```
[SENSe]:NOISe:FIGure:POINTs:COUNt { P801 | P2401 | P4001 | P8001 | P10401 }
```

Arguments

P801 sets the number of sample points to 801.

P2401 sets the number of sample points to 2401.

P4001 sets the number of sample points to 4001.

P8001 sets the number of sample points to 8001.

P10401 sets the number of sample points to 10401.

Examples

SENSe:NOISE:FIGURE:POINTS:COUNT P801 sets the number of sample points to 801.

[SENSe]:NOISe:FREQuency:CENTer

Sets or queries the center frequency value (Hz).

Conditions

Measurement views: Noise Figure, Gain, Noise Temperature, Y Factor, Noise Table

Group

Sense commands

Syntax

```
[SENSe]:NOISe:FREQuency:CENTer <NRf>
[SENSe]:NOISe:FREQuency:CENTer?
```

Arguments

<NRf> is the center frequency value in Hz.

Range: 0 Hz to 6.2 GHz

Returns

<NR3> = floating point value with an exponent, which is the center frequency value.

Examples

SENSe:NOISE:FREQUENCY:CENTER 5.0 GHz sets the center frequency in the active noise display to 5.0000 GHz.

SENSe:NOISE:FREQUENCY:CENTER? might return 6.2000000000E+9, indicating that the center frequency in the active noise display is 6.2000 GHz.

[SENSe]:NOISe:FREQuency:MODE

Sets or queries the frequency mode in the noise display.

Conditions

Measurement views: Noise Figure, Gain, Noise Temperature, Y Factor, Noise Table

Group

Sense commands

Syntax

```
[SENSe]:NOISe:FREQuency:MODE { TABLe | SINGle | SWEPt }
[SENSe]:NOISe:FREQuency:MODE?
```

Arguments

TABLe: sets the frequency mode to Frequency Table.

SINGle: sets the frequency mode to Single Frequency.

SWEPt: sets the frequency mode to Sweep Frequency.

Returns

TABL: means the frequency mode is set to Frequency Table.

SING: means the frequency mode is set to Single Frequency.

SWEP: means the frequency mode is set to Sweep Frequency.

Examples

SENSe:NOIS:FREQ:MODE TABL sets the frequency mode to Frequency Table in the noise display.

[SENSe]:NOISe:FREQuency:SPAN

Sets or queries the frequency Span value (Hz).

Conditions

Measurement views: Noise Figure, Gain, Noise Temperature, Y Factor, Noise Table

Group

Sense commands

Syntax

```
[SENSe]:NOISe:FREQuency:SPAN <NRf>
[SENSe]:NOISe:FREQuency:SPAN?
```

Arguments

<NRf> is the frequency Span value in Hz.

Range: 100 Hz to 6.2 GHz

Returns

<NR3> = floating point value with an exponent, which is the frequency Span value.

Examples

SENSe:NOISE:FREQUENCY:SPAN 3.0 GHz sets the frequency Span in the active noise display to 3.0000 GHz.

SENSe:NOISE:FREQUENCY:SPAN? might return 100.000000000E+3, indicating that the frequency Span in the active noise display is 100 kHz.

[SENSe]:NOISe:FREQuency:STARt

Sets or queries the Start Frequency value.

Conditions

Measurement views: Noise Figure, Gain, Noise Temperature, Y Factor, Noise Table

Group

Sense commands

Syntax

```
[SENSe]:NOISe:FREQuency:STARt <NRf>
[SENSe]:NOISe:FREQuency:STARt?
```

Arguments

<NRf> is the Start Frequency value.

Returns

<NR3> = floating point value with an exponent, which is the Start Frequency value.

Examples

SENSe:NOISE:FREQUENCY:START 1.5 GHz sets the Start Frequency in the active noise display to 1.5000 GHz.

SENSe:NOISE:FREQUENCY:START? might return 1.5000000000E+9, indicating that the Start Frequency in the active noise display is 1.5000 GHz.

[SENSe]:NOISe:FREQuency:STEPs

Sets or queries the frequency Steps value.

Conditions

Measurement views: Noise Figure, Gain, Noise Temperature, Y Factor, Noise Table

Group

Sense commands

Syntax

```
[SENSe]:NOISe:FREQuency:STEPs <NRf>
[SENSe]:NOISe:FREQuency:STEPs?
```

Arguments

<NRf> is the Steps value.

Range: 2 to 999

Returns

<NRf> is the Steps value.

Examples

SENSe:NOISE:FREQUENCY:STEPS 11 sets the frequency Steps value in the active noise display to 11.

[SENSe]:NOISe:FREQuency:STOP

Sets or queries the Stop Frequency value.

Conditions

Measurement views: Noise Figure, Gain, Noise Temperature, Y Factor, Noise Table

Group

Sense commands

Syntax

```
[SENSe]:NOISe:FREQuency:STOP <NRf>
[SENSe]:NOISe:FREQuency:STOP?
```

Arguments

<NRf> is the Stop Frequency value.

Returns

<NR3> = floating point value with an exponent, which is the Stop Frequency value.

Examples

SENSe:NOISE:FREQUENCY:STOP 1.5 GHz sets the Stop Frequency in the active noise display to 1.5000 GHz.

SENSe:NOISE:FREQUENCY:STOP? might return 1.5000000000E+9, indicating that the Stop Frequency in the active noise display is 1.5000 GHz.

[SENSe]:NOISe:FREQuency:TABLe:DATA? (Query Only)

Queries the data in the Frequency Table for the noise display.

Conditions

Measurement views: Noise Figure, Gain, Noise Temperature, Y Factor, Noise Table

Group

Sense commands

Syntax

```
[SENSe]:NOISe:FREQuency:TABLe:DATA?
```

Returns

<string> which contains each frequency value (Hz) in the frequency table for the noise display.

Examples

SENSe:NOISE:FREQUENCY:TABLE:DATA? might return "1500000000, 1505000005, 1510000010, 1515000015,1520000020, 1525000025, 1530000030, 1545000045, 1550000050", indicating each frequency in the Frequency Table.

[SENSe]:NOISe:FREQuency:TABLe:NEW (No Query Form)

Creates a new Frequency Table with the specified values for the noise display.

Conditions

Measurement views: Noise Figure, Gain, Noise Temperature, Y Factor, Noise Table

Group

Sense commands

Syntax

```
[SENSe]:NOISe:FREQuency:TABLe:NEW <string>
```

Arguments

```
<string> = "<NR3>,<NR3>,<NR3>,..."
```

Examples

SENSe:NOISE:FREQUENCY:TABLE:NEW "1.2e9,1.5e9,1.452e9" sets 3 step values in a new Frequency Table to 1.200 GHz, 1.350 GHz, and 1.500 GHz.

[SENSe]:OBWidth:AVERage

Sets or queries whether to enable averaging in the Occupied Bandwidth measurement.

Conditions

Measurement views: Occupied Bandwidth

Group

Sense commands

Syntax

```
[SENSe]:OBWidth:AVERage { OFF | ON | 0 | 1 }
[SENSe]:OBWidth:AVERage?
```

Arguments

OFF or 0 disables averaging.

ON or 1 enables averaging.

Examples

SENSe:OBWIDTH:AVERAGE ON enables averaging.

[SENSe]:OBWidth:AVERage:COUNt

Sets or queries the number of measurements for averaging in the Occupied Bandwidth measurement.

Conditions

Measurement views: Occupied Bandwidth

Group

Sense commands

Syntax

```
[SENSe]:OBWidth:AVERage:COUNt <number>
[SENSe]:OBWidth:AVERage:COUNt?
```

Arguments

`<number>::<NR1>` specifies the average count. Range: 2 to 10000.

Examples

`SENSe:OBWIDTH:AVERAGE:COUNT 64` sets the average count to 64.

[SENSe]:OBWidth:{BANDwidth|BWIDth}:MEASurement

Sets or queries the measurement bandwidth to determine the total power in the Occupied Bandwidth measurement.

Conditions

Measurement views: Occupied Bandwidth

Group

Sense commands

Syntax

```
[SENSe]:OBWidth:{BANDwidth|BWIDth}:MEASurement <value>
[SENSe]:OBWidth:{BANDwidth|BWIDth}:MEASurement?
```

Arguments

`<value>::<NRf>` specifies the measurement bandwidth.

Range: 100 Hz to 109 MHz.

Examples

`SENSe:OBWIDTH:BANDWIDTH:MEASUREMENT 10MHz` sets the measurement bandwidth to 10 MHz.

[SENSe]:OBWidth:{BANDwidth|BWIDth}[:RESolution]

Sets or queries the resolution bandwidth (RBW). Programming a specified RBW sets `[SENSe]:OBWidth{BANDwidth|BWIDth}[:RESolution]:AUTO OFF`.

Conditions

Measurement views: Occupied Bandwidth

Group

Sense commands

Syntax

```
[SENSe]:OBWidth:{BANDwidth|BWIDth}[:RESolution] <value>
[SENSe]:OBWidth:{BANDwidth|BWIDth}[:RESolution]?
```

Related Commands

[SENSe]:OBWidth:{BANDwidth|BWIDth}[:RESolution]:AUTO (on page 2206)

Arguments

`<value>::<NRf>` specifies the RBW. Range: 100 Hz to 5 MHz.

Examples

`SENSe:OBWIDTH:BANDWIDTH:RESOLUTION 200kHz` sets the RBW to 200 kHz.

[SENSe]:OBWidth:{BANDwidth|BWIDth}[:RESolution]:ACTual? (Query Only)

Queries the actual resolution bandwidth (RBW) in the Occupied Bandwidth measurement.

Conditions

Measurement views: Occupied Bandwidth

Group

Sense commands

Syntax

```
[SENSe]:OBWidth:{BANDwidth|BWIDth}[:RESolution]:ACTual?
```

Arguments

None

Returns

`<NRf>` The actual RBW in Hz.

Examples

`SENSe:OBWIDTH:BANDWIDTH:RESOLUTION:ACTUAL?` might return `299.624E+3`, indicating that the actual RBW is 299.624 kHz.

[SENSe]:OBWidth:{BANDwidth|BWIDth}[:RESolution]:AUTO

Determines whether to set the resolution bandwidth (RBW) automatically or manually in the Occupied Bandwidth measurement.

Conditions

Measurement views: Occupied Bandwidth

Group

Sense commands

Syntax

```
[SENSe]:OBWidth:{BANDwidth|BWIDth}[:RESolution]:AUTO { OFF | ON | 0 | 1 }
[SENSe]:OBWidth:{BANDwidth|BWIDth}[:RESolution]:AUTO?
```

Arguments

OFF or 0 specifies that the RBW is set manually using the [SENSe]:OBWidth:{BANDwidth|BWIDth}[:RESolution] command.

ON or 1 specifies that the RBW is set automatically.

Examples

SENSe:OBWIDTH:BANDWIDTH:AUTO ON sets the RBW automatically.

[SENSe]:OBWidth:{BANDwidth|BWIDth}:VIDeo

Sets or queries the video bandwidth (VBW). Programming a specified VBW sets [SENSe]:OBWidth{BANDwidth|BWIDth}:VIDeo:STATe OFF.

Conditions

Measurement views: Occupied Bandwidth

Group

Sense commands

Syntax

```
[SENSe]:OBWidth:{BANDwidth|BWIDth}:VIDeo <value>
[SENSe]:OBWidth:{BANDwidth|BWIDth}:VIDeo?
```

Arguments

<value>::<NRf> specifies the VBW.

Range: Current RBW/104 (1 Hz minimum) to Current RBW.

Examples

SENSe:OBWIDTH:BANDWIDTH:VIDEO 200kHz sets the VBW to 200 kHz.

[SENSe]:OBWidth:{BANDwidth|BWIDth}:VIDeo:STATe

Determines whether to enable the video bandwidth (VBW) in the Occupied Bandwidth measurement.

Conditions

Measurement views: Occupied Bandwidth

Group

Sense commands

Syntax

```
[SENSe]:OBWidth:{BANDwidth|BWIDth}:VIDeo:STATe { OFF | ON| 0 | 1 }
[SENSe]:OBWidth:{BANDwidth|BWIDth}:VIDeo:STATe?
```

Arguments

OFF or 0 disables the VBW.

ON or 1 enables the VBW.

Examples

SENSe:OBWIDTH:BANDWIDTH:VIDEO:STATE ON enables the VBW.

[SENSe]:OBWidth:BOBW:XDBM:MEASurement:DIREction

Sets or queries the measurement direction in the Bluetooth window.

Conditions

Measurement views:Bluetooth 20 dB Bandwidth

Group

Sense commands

Syntax

```
[SENSe]:OBWidth:BOBW:XDBM:MEASurement:DIREction {IN|OUT}
[SENSe]:OBWidth:BOBW:XDBM:MEASurement:DIREction?
```

Arguments

`IN` specifies the inward direction

`OUT` specifies the outward direction.

Examples

`SENSe:OBWIDTH:BOBW:XDBM:MEASUREMENT:DIRECTION?` might return `IN`, indicating the measurement direction is set to IN.

[SENSe]:OBWidth:BOBW:XDBM:RANGe

Sets or queries the measurement range in the Bluetooth window.

Conditions

Measurement views:Bluetooth 20 dB Bandwidth

The x dBm option should be selected. If the x dB option is selected, the Range is always set to BOTH.

Group

Sense commands

Syntax

```
[SENSe]:OBWidth:BOBW:XDBM:RANGe {BOTH|LOW|HIGH}
[SENSe]:OBWidth:BOBW:XDBM:RANGe?
```

Arguments

`BOTH` specifies both the high and low ranges.

`LOW` specifies the lower range.

`HIGH` specifies the higher range.

Examples

`SENSe:OBWIDTH:BOBW:XDBM:RANGE LOW` sets the measurement range to the lower range.

[SENSe]:OBWidth:CLEar:RESults (No Query Form)

Restarts the average trace. This command is valid when `[SENSe]:OBWidth:AVERage` is set to ON.

Conditions

Measurement views: Occupied Bandwidth

Group

Sense commands

Syntax

```
[SENSe]:OBWidth:CLEar:RESults
```

Arguments

None

Examples

`SENSe:OBWIDTH:CLEAR:RESULTS` restarts the average trace.

[SENSe]:OBWidth:FREQuency:CENTer

Sets or queries the center frequency in the Occupied Bandwidth measurement.

Conditions

Measurement views: Occupied Bandwidth

Group

Sense commands

Syntax

```
[SENSe]:OBWidth:FREQuency:CENTer <value>
[SENSe]:OBWidth:FREQuency:CENTer?
```

Arguments

`<value>::<NRf>` specifies the center frequency.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples

`SENSe:OBWIDTH:FREQUENCY:CENTER 7.5GHz` sets the center frequency to 7.5 GHz.

[SENSe]:OBWidth:FREQuency:STEP

Sets or queries the frequency step size in the Occupied Bandwidth measurement. Programming a specified step size sets `[SENSe]:OBWidth:FREQuency:STEP:AUTO OFF`.

Conditions

Measurement views: Occupied Bandwidth

Group

Sense commands

Syntax

```
[SENSe]:OBWidth:FREQuency:STEP <value>
[SENSe]:OBWidth:FREQuency:STEP?
```

Related Commands

[SENSe]:OBWidth:FREQuency:STEP:AUTO (on page 2210)

Arguments

`<value>::<NRf>` specifies the frequency step size.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples

`SENSe:OBWIDTH:FREQUENCY:STEP 1kHz` sets the frequency step size to 1 kHz.

[SENSe]:OBWidth:FREQuency:STEP:AUTO

Determines whether to set the frequency step size automatically or manually in the Occupied Bandwidth measurement.

Conditions

Measurement views: Occupied Bandwidth

Group

Sense commands

Syntax

```
[SENSe]:OBWidth:FREQuency:STEP:AUTO{OFF|ON|0|1}
[SENSe]:OBWidth:FREQuency:STEP:AUTO?
```

Arguments

`OFF` or `0` specifies that the frequency step size is set manually using the `[SENSe]:OBWidth:FREQuency:STEP` command.

`ON` or `1` specifies that the frequency step size is set automatically.

Examples

`SENSe:OBWIDTH:FREQUENCY:STEP:AUTO ON` specifies that the frequency step size is set automatically.

[SENSe]:OBWidth:PERCent

Sets or queries the occupied bandwidth percent power (power ratio of the occupied bandwidth to the measurement bandwidth).

Conditions

Measurement views: Occupied Bandwidth

Group

Sense commands

Syntax

```
[SENSe]:OBWidth:PERCent <value>
[SENSe]:OBWidth:PERCent?
```

Arguments

`<value>::<NRf>` specifies the occupied bandwidth percent power.

Range: 50 to 99.9%.

Examples

`SENSe:OBWIDTH:PERCENT 98` sets the occupied bandwidth percent power to 98%.

[SENSe]:OBWidth:XDBLevel

Sets or queries the x dB level (how far down from the peak level the bandwidth is measured) in the x dB bandwidth measurement.

Conditions

Measurement views: Occupied Bandwidth

Group

Sense commands

Syntax

```
[SENSe]:OBWidth:XDBLevel <value>
[SENSe]:OBWidth:XDBLevel?
```

Arguments

`<value>::<NRf>` specifies the x dB level. Range: -80 to -1 dB.

Examples

`SENSe:OBWIDTH:XDBLEVEL -10` sets the x dB level to -10 dB.

[SENSe]:OFDM:CHANnel[:BANDwidth|BWIDth]

Sets or queries the channel bandwidth in the OFDM view.

Conditions

Measurement views: OFDM

Group

Sense commands

Syntax

```
[SENSe]:OFDM:CHANnel[:BANDwidth|BWIDth] <value>
[SENSe]:OFDM:CHANnel[:BANDwidth|BWIDth]?
```

Arguments

`<value>::<NRf>` specifies the channel bandwidth.

Examples

`SENSe:OFDM:CHANNEL[:BANDWIDTH|BWIDTH] 20E+6` sets the channel bandwidth to 20 MHz.

[SENSe]:OFDM:CHANnel[:BANDwidth|BWIDth]:SRATe? (Query Only)

Returns the FFT sample rate for the OFDM view.

Conditions

Measurement views: OFDM

Group

Sense commands

Syntax

`[SENSe]:OFDM:CHANnel[:BANDwidth|BWIDth]:SRATe?`

Arguments

None

Returns

`<NRf>` The sample rate in Hz.

Examples

`SENSe:OFDM:CHANNEL[:BANDWIDTH|BWIDTH]:SRATE?` might return `20E+6`, indicating the sample rate is 20 MHz.

[SENSe]:OFDM:CHANnel[:BANDwidth|BWIDth]:STATe

Determines whether the channel bandwidth is on or off in the OFDM view.

Conditions

Measurement views: OFDM

Group

Sense commands

Syntax

```
[SENSe]:OFDM:CHANnel[:BANDwidth|BWIDth]:STATe{OFF|ON|0|1}
[SENSe]:OFDM:CHANnel[:BANDwidth|BWIDth]:STATe?
```

Arguments

`OFF` or `0` specifies that the channel bandwidth is off.

`ON` or `1` specifies that the channel bandwidth is on.

Examples

`SENSe:OFDM:CHANNEL[:BANDWIDTH|BWIDTH]:STATE ON` sets the channel bandwidth setting to on.

[SENSe]:OFDM:CHANnel:ESTimation

Sets or queries the channel estimation in the OFDM view.

Conditions

Measurement views: OFDM

Group

Sense commands

Syntax

```
[SENSe]:OFDM:CHANnel:ESTimation{PREamble|PDATa}
[SENSe]:OFDM:CHANnel:ESTimation?
```

Arguments

`PREamble` specifies the channel estimation is Preamble.

`PDATa` specifies the channel estimation is Preamble + Data.

Examples

`SENSe:OFDM:CHANNEL:ESTIMATION PDAT` sets the channel estimation to Preamble + Data.

[SENSe]:OFDM:CONSte:DETermination

Sets or queries the constellation determination in the OFDM view.

Conditions

Measurement views: OFDM

Group

Sense commands

Syntax

```
[SENSe]:OFDM:CONSte:DETermination{AUTO|MANual}
[SENSe]:OFDM:CONSte:DETermination?
```

Arguments

`AUTO` specifies the instrument automatically detects the constellation determination.

`MANual` allows you to select a manual constellation type.

Examples

`SENSe:OFDM:CONSTE:DETERMINATION MAN` sets the constellation determination so you can select a manual constellation type using the `SENSe:OFDM:CONSte:DETermination:MANual` command.

[SENSe]:OFDM:CONSte:DETermination:MANual

Sets or queries a manual constellation type. The Constellation determination must be set to Manual using the `[SENSe]:OFDM:CONSte:DETermination` command.

Conditions

Measurement views: OFDM

Group

Sense commands

Syntax

```
[SENSe]:OFDM:CONSte:DETermination:MANual{BPSK|QAM16|QAM64|QPSK}
[SENSe]:OFDM:CONSte:DETermination:MANual?
```

Related Commands

[SENSe]:OFDM:CONSte:DETermination (on page 2213)

Arguments

`BPSK` specifies the BPSKconstellation type.

`QAM16` specifies the QAM16 constellation type.

`QAM64` specifies the QAM16 constellation type.

`QPSK` specifies the QPSK constellation type.

Examples

`SENSe:OFDM:CONSTE:DETERMINATION:MANUAL QPSK` selects the QPSK constellation type.

[SENSe]:OFDM:GUARd:INTerval

Sets or queries the guard interval in the OFDM view.

Conditions

Measurement views: OFDM

Group

Sense commands

Syntax

```
[SENSe]:OFDM:GUARd:INTerval{FOURth|EIGHth|SIXTeenth|THIRtysecond|USER}
```

Arguments

`FOURth` specifies the guard interval is set to 1/4.

`EIGHth` specifies the guard interval is set to 1/8.

`SIXTeenth` specifies the guard interval is set to 1/16.

`THIRtysecond` specifies the guard interval is set to 1/32.

`USER` specifies a user-defined guard interval.

Examples

`SENSe:OFDM:GUARD:INTERVAL THIR` sets the guard interval to 1/32.

[SENSe]:OFDM:GUARd:INTerval:USER

Sets or queries a user-defined guard interval. The Guard Interval must be set to USER.

Conditions

Measurement views: OFDM

Group

Sense commands

Syntax

```
[SENSe]:OFDM:GUARd:INTerval:USER <value>
[SENSe]:OFDM:GUARd:INTerval:USER?
```

Related Commands

[SENSe]:OFDM:GUARd:INTerval (on page 2214)

Arguments

`<value>::<NRf>` specifies the guard interval.

Examples

`SENSe:OFDM:GUARD:INTERVAL:USER 3.12` sets the guard interval to 3.12%.

[SENSe]:OFDM:PILot:TRACking:AMPLitude:STATe

Determines whether to select the Amplitude pilot tracking in the OFDM view.

Conditions

Measurement views: OFDM

Group

Sense commands

Syntax

```
[SENSe]:OFDM:PILot:TRACking:AMPLitude:STATe { OFF | ON | 0 | 1 }
[SENSe]:OFDM:PILot:TRACking:AMPLitude:STATe?
```

Arguments

`OFF` or `0` specifies that the Amplitude pilot tracking is off.

`ON` or `1` specifies that the Amplitude pilot tracking is on.

Examples

`SENSe:OFDM:PILOT:TRACKING:AMPLITUDE:STATE ON` turns the Amplitude pilot tracking on.

[SENSe]:OFDM:PILot:TRACking:PHASe:STATe

Determines whether to select the Phase pilot tracking in the OFDM view.

Conditions

Measurement views: OFDM

Group

Sense commands

Syntax

```
[SENSe]:OFDM:PILot:TRACking:PHASe:STATe {OFF|ON|0|1}
```

Arguments

`OFF` or `0` specifies that the Phase pilot tracking is off.

`ON` or `1` specifies that the Phase pilot tracking is on.

Examples

`SENSe:OFDM:PILOT:TRACKING:PHASE:STATE ON` turns the Phase pilot tracking on.

[SENSe]:OFDM:PILot:TRACking:TIMing:STATe

Determines whether to select the Timing pilot tracking in the OFDM view.

Conditions

Measurement views: OFDM

Group

Sense commands

Syntax

```
[SENSe]:OFDM:PILot:TRACking:TIMing:STATe {OFF|ON|0|1}
```

Arguments

`OFF` or `0` specifies that the Timing pilot tracking is off.

`ON` or `1` specifies that the Timing pilot tracking is on.

Examples

`SENSe:OFDM:PILOT:TRACKING:TIMING:STATE ON` turns the Timing pilot tracking on.

[SENSe]:OFDM:RADix (No Query Form)

Sets the radix for the OFDM measurements. This command is effective in the symbol table.

Conditions

Measurement views: OFDM

Group

Sense commands

Syntax

```
[SENSe]:OFDM:RADix { BINary | HEX }
```

Arguments

Returns

`BINary` specifies the binary radix.

`HEX` specifies the hexadecimal radix.

Examples

SENSe:OFDM:RADIX HEX sets the radix to hexadecimal for the OFDM symbol table.

[SENSe]:OFDM:SCARrier:SPACing

Sets or queries the subcarrier spacing in the OFDM view.

Conditions

Measurement views: OFDM

Group

Sense commands

Syntax

```
[SENSe]:OFDM:SCARrier:SPACing <value>
[SENSe]:OFDM:SCARrier:SPACing?
```

Arguments

<value>::<NRf> specifies the subcarrier spacing.

Examples

SENSe:OFDM:SCARRIER:SPACING 312.5E+3 sets the subcarrier spacing to 312.5 MHz.

[SENSe]:OFDM:SCARrier:SPACing:STATe

Determines whether the subcarrier spacing is on or off in the OFDM view

Conditions

Measurement views: OFDM

Group

Sense commands

Syntax

```
[SENSe]:OFDM:SCARrier:SPACing:STATe{OFF|ON|0|1}
[SENSe]:OFDM:SCARrier:SPACing:STATe?
```

Arguments

OFF or 0 specifies that the subcarrier spacing is off.

ON or 1 specifies that the subcarrier spacing is on.

Examples

SENSe:OFDM:SCARRIER:SPACING:STATE ON sets the subcarrier spacing to on.

[SENSe]:OFDM:SCARriers

Sets or queries the Subcarriers to display in the OFDM view.

Conditions

Measurement views: OFDM

Group

Sense commands

Syntax

```
[SENSe]:OFDM:SCARriers { ALL | PILot | DATA | RANGe | SINGle}
[SENSe]:OFDM:SCARriers?
```

Arguments

`ALL` specifies that all subcarriers will be displayed.

`PILot` specifies that only the pilot subcarriers will be displayed.

`DATA` specifies that only the data subcarriers will be displayed.

`RANGe` specifies that the subcarriers within a specified range will be displayed.

`SINGle` specifies that only a single subcarrier will be displayed.

Examples

`SENSe:OFDM:SCARRIERS ALL` sets the OFDM view to display all subcarriers.

[SENSe]:OFDM:SCARriers:RANGe:STARt

Sets or queries the subcarrier start range in the OFDM view.

Conditions

Measurement views: OFDM

Group

Sense commands

Syntax

```
[SENSe]:OFDM:SCARriers:RANGe:STARt <value>
[SENSe]:OFDM:SCARriers:RANGe:STARt?
```

Related Commands

[SENSe]:OFDM:SCARriers:RANGe:STOP (on page 2218)

Arguments

`<NR1>`

Examples

`SENSe:OFDM:SCARRIERS:RANGE:START -20` sets the subcarrier start range to –20.

[SENSe]:OFDM:SCARriers:RANGe:STOP

Sets or queries the subcarrier stop range in the OFDM view.

Conditions

Measurement views: OFDM

Group

Sense commands

Syntax

```
[SENSe]:OFDM:SCARriers:RANGe:STOP <value>
[SENSe]:OFDM:SCARriers:RANGe:STOP?
```

Related Commands

[SENSe]:OFDM:SCARriers:RANGe:STARt (on page 2218)

Arguments

<NR1>

Examples

SENSe:OFDM:SCARRIERS:RANGE:STOP 20 sets the subcarrier stop range to 20.

[SENSe]:OFDM:SCARriers:SINGle:INDex

Sets or queries the index of a single subcarrier in the OFDM view.

Conditions

Measurement views: OFDM

Group

Sense commands

Syntax

```
[SENSe]:OFDM:SCARriers:SINGle:INDex <Value>
[SENSe]:OFDM:SCARriers:SINGle:INDex?
```

Arguments

<NR1>

Examples

SENSe:OFDM:SCARRIERS:SINGLE:INDEX -10 sets the single subcarrier index to −10.

[SENSe]:OFDM:FFT:LENGth? (Query Only)

Returns the FFT length for the OFDM view.

Conditions

Measurement views: OFDM

Group

Sense commands

Syntax

```
[SENSe]:OFDM:FFT:LENGth?
```

Arguments

None

Returns

<NR1> The FFT length.

Examples

SENSe:OFDM:FFT:LENGTH? might return the FFT length as 64.

[SENSe]:OFDM:STANdard

Sets or queries the OFDM standard.

Conditions

Measurement views: OFDM

Group

Sense commands

Syntax

```
[SENSe]:OFDM:STANdard { S802_11 | S802_16 }
[SENSe]:OFDM:STANdard?
```

Arguments

S802_11 specifies the 802.11a/g/j OFDM standard.

S802_16 specifies the 802.16 (2004) standard.

Examples

SENSe:OFDM:STANDARD S802_16 sets the OFDM standard to 802.16.

[SENSe]:OFDM:SWAP:IQ

Determines whether or not to exchange I and Q data before demodulating.

Conditions

Measurement views: OFDM

Group

Sense commands

Syntax

```
[SENSe]:OFDM:SWAP:IQ{OFF|ON|0|1}
[SENSe]:OFDM:SWAP:IQ?
```

Arguments

OFF or 0 uses I and Q data as they are.

ON or 1 exchanges I and Q data.

Examples

SENSe:OFDM:SWAP:IQ ON exchanges I and Q data before demodulating.

[SENSe]:OFDM:SYMBol:ANALysis:OFFSet

Sets or queries the OFDM symbol analysis offset.

Conditions

Measurement views: OFDM

Group

Sense commands

Syntax

```
[SENSe]:OFDM:SYMBol:ANALysis:OFFSet <value>
[SENSe]:OFDM:SYMBol:ANALysis:OFFSet?
```

Arguments

`<value>` specifies the symbol analysis offset as a percentage.

Range 0 to −100%.

Examples

`SENSe:OFDM:SYMBOL:ANALYSIS:OFFSET -50` sets the symbol analysis offset to −50%.

[SENSe]:OFDM:SYMBols

Sets or queries the symbols to display in the OFDM view.

Conditions

Measurement views: OFDM

Group

Sense commands

Syntax

```
[SENSe]:OFDM:SYMBols{ALL|RANGe|SINGle}
[SENSe]:OFDM:SYMBols?
```

Arguments

`ALL` specifies that all symbols will be displayed.

`RANGe` specifies that a range of symbols will be displayed.

`SINGle` specifies that single symbol will be displayed.

Examples

`SENSe:OFDM:SYMBOLS ALL` specifies that all symbols will be displayed in the OFDM views.

[SENSe]:OFDM:SYMBols:MAX

Sets or queries the maximum number of symbols to analyze in the OFDM views.

Conditions

Measurement views: OFDM

Group

Sense commands

Syntax

```
[SENSe]:OFDM:SYMBols:MAX <value>
[SENSe]:OFDM:SYMBols:MAX?
```

Arguments

<NR1>

Examples

SENSe:OFDM:SYMBOLS:MAX 50 specifies a maximum of 50 symbols to analyze in the OFDM views.

[SENSe]:OFDM:SYMBols:MAX:STATe

Enables or disables the maximum number of symbols to analyze in the OFDM view.

Conditions

Measurement views: OFDM

Group

Sense commands

Syntax

```
[SENSe]:OFDM:SYMBols:MAX:STATe{OFF|ON|0|1}
[SENSe]:OFDM:SYMBols:MAX:STATe?
```

Related Commands

[SENSe]:OFDM:SYMBols:MAX (on page 2221)

Arguments

OFF or 0 disables the maximum number of symbols to analyze.

ON or 1 enables the maximum number of symbols to analyze.

Examples

SENSe:OFDM:SYMBOLS:MAX:STATE ON enables the number of symbols to analyze in the OFDM view. The number is specified by the [SENSe]:OFDM:SYMBols:MAX command.

[SENSe]:OFDM:SYMBols:RANGe:COUNt

Sets or queries the number of symbols to display in the OFDM view when displaying a range of symbols.

Conditions

Measurement views: OFDM

Group

Sense commands

Syntax

```
[SENSe]:OFDM:SYMBols:RANGe:COUNt<value>
[SENSe]:OFDM:SYMBols:RANGe:COUNt?
```

Arguments

<NR1>

Examples

SENSe:OFDM:SYMBOLS:RANGE:COUNT 25 sets the number of symbols to display to 25.

[SENSe]:OFDM:SYMBols:RANGe:STARt

Sets or queries the symbols start range in the OFDM view when displaying a range of symbols.

Conditions

Measurement views: OFDM

Group

Sense commands

Syntax

```
[SENSe]:OFDM:SYMBols:RANGe:STARt <value>
[SENSe]:OFDM:SYMBols:RANGe:STARt?
```

Arguments

<NR1>

Examples

SENSe:OFDM:SYMBOLS:RANGE:START 0 sets the start range to 0.

[SENSe]:OFDM:SYMBols:SINGle:INDex

Sets or queries the symbol index in the OFDM view when displaying a single symbol.

Conditions

Measurement views: OFDM

Group

Sense commands

Syntax

```
[SENSe]:OFDM:SYMBols:SINGle:INDex <value>
[SENSe]:OFDM:SYMBols:SINGle:INDex?
```

Arguments

<NR1>

Examples

SENSe:OFDM:SYMBOLS:SINGLE:INDEX 0 sets the index to 0.

[SENSe]:OFDM:UNIT:FREQuency (No Query Form)

Specifies the frequency units for the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Sense commands

Syntax

```
[SENSe]:OFDM:UNIT:FREQuency{FREQuency|SCARrier}
```

Arguments

`FREQuency` sets the frequency units to frequency.

`SCARrier` sets the frequency units to subcarriers.

Examples

`SENSe:OFDM:UNIT:FREQUENCY FREQ` sets the frequency units to frequency.

[SENSe]:OFDM:UNIT:TIME (No Query Form)

Specifies the time units for the OFDM measurement.

Conditions

Measurement views: OFDM

Group

Sense commands

Syntax

```
[SENSe]:OFDM:UNIT:TIME{SEConds|SYMBols}
```

Arguments

`SEConds` sets the time units to seconds.

`SYMBols` sets the time units to symbols.

Examples

`SENSe:OFDM:UNIT:TIME SEC` sets the time units to seconds.

[SENSe]:P25:ANALysis:OFFSet

Sets or queries the Analysis Offset value from the P25 displays.

Conditions

Measurement views: P25 displays

Group

Sense commands

Syntax

```
[SENSe]:P25:ANALysis:OFFSet <value>
```

Arguments

`<NRf>`

Examples

`SENSe:P25:ANALYSIS:OFFSET 2` sets the Analysis Offset value to 2.

[SENSe]:P25:ANALysis:OFFSet:AUTO

Sets or queries the Analysis Offset auto checkbox selection from P25 display.

Conditions

Measurement views: P25 displays

Group

Sense commands

Syntax

```
[SENSe]:P25:ANALysis:OFFSet:AUTO <value>
```

Arguments

`0` unchecks the checkbox.

`1` checks the checkbox.

Returns

0 or 1, `<NRf>`

Examples

`SENSe:P25:ANALYSIS:OFFSET:AUTO 1` enables the Analysis Offset auto checkbox in P25 display.

[SENSe]:P25([:BANDwidth]|:BWIDth):MODE

Sets or queries the Measurement BW (bandwidth) (frequency span) mode as Auto, Manual, or Link to Span.

Conditions

Measurement view: Any P25 measurement display

Group

Sense commands

Syntax

```
[SENSe]:P25([:BANDwidth]|:BWIDth):MODE {MANual|AUTO|LSP}
[SENSe]:P25([:BANDwidth]|:BWIDth):MODE?
```

Arguments

`MANual` sets the analyzer to allow for manual selection of the measurement bandwidth. A typical use is setting a narrow measurement bandwidth on a small range of frequencies to examine one signal while rejecting the others that are present in a Spectrum graph.

`AUTO` sets analyzer to automatically select a measurement bandwidth. The measurement picks a bandwidth based on other parameter settings (such as symbol rate and modulation type).

`LSP` sets the Measurement BW to Link to Span. This allows you to use the Spectrum display to tune the frequency, view the signal, and then use the Span control to set the measurement bandwidth. This mode emulates legacy instruments.

Returns

MAN means the Measurement BW is set to Manual.

AUTO means the Measurement BW is set to Automatic.

LSP means the Measurement BW is set to Link to Span.

Examples

`SENSe:P25:BAND:MODE?` might return `MAN`, indicating the Measurement BW mode is set to AUTO.

[SENSe]:P25([:BANDwidth]|:BWIDth):TINTerval

Sets or queries the Measurement BW (bandwidth) (frequency span). Setting the Measurement BW changes this setting from Auto to Manual if it is set to Auto before you use the command. A query will not change that setting.

Conditions

Measurement view: Any P25 measurement display

Group

Sense commands

Syntax

```
[SENSe]:P25([:BANDwidth]|:BWIDth):TINTerval <NRf>
[SENSe]:P25([:BANDwidth]|:BWIDth):TINTerval?
```

Arguments

`<NRf>` specifies the measurement bandwidth in kHz.

Returns

`<NR3>` = a floating point value with an exponent, which is the Measurement BW value in kHz.

Examples

`SENSe:P25:BAND:INT?` might return `153.6000000000E+3`, which indicates the Measurement BW value is 153.6 kHz.

[SENSe]:P25([:BANDwidth]|:BWIDth):TINTerval:AUTO

Sets or queries the Measurement BW (bandwidth) (frequency span) mode as Auto or Manual.

Conditions

Measurement view: Any P25 measurement display

Group

Sense commands

Syntax

```
[SENSe]:P25([:BANDwidth]|:BWIDth):TINTerval:AUTO {0|1|OFF|ON}
[SENSe]:P25([:BANDwidth]|:BWIDth):TINTerval:AUTO?
```

Arguments

`OFF` or `0` sets Measurement BW to Auto.

`ON` or `1` sets Measurement BW to Manual.

Returns

0 means Measurement BW is set to Auto.

1 means Measurement BW is set to Manual.

Examples

`SENSe:P25:BAND:INT:AUTO?` might return `1`, indicating the Measurement BW is set to Auto.

[SENSe]:P25:FILTer:MEASurement

Sets or queries the P25 measurements Measurement Filter setting. The User Meas Filter selections are predefined user filters.

Conditions

Measurement view: Any P25 measurement display

Group

Sense commands

Syntax

```
[SENSe]:P25:FILTer:MEASurement{ OFF | USER1 | USER2 | USER3 | UOTH | RECT | RRC | RCOS | GAUS |
   HDQPSK_P25 | C4FM_P25}
[SENSe]:P25:FILTer:MEASurement?
```

Arguments

`OFF` sets the filter to None.

`USER1` sets the filter to User Meas Filter 1.

`USER2` sets the filter to User Meas Filter 2.

`USER3` sets the filter to User Meas Filter 3.

`UOTH` sets the filter to User other.

`RECT` sets the filter to Rectangular (only available for Phase 2 HDQPSK modulation type).

`RRC` sets the filter to Root Raised Cosine (only available for Phase 2 HDQPSK modulation type).

`RCOS` sets the filter to Raised Cosine (only available for Phase 2 HDQPSK modulation type).

`GAUS` sets the filter to Gaussian (only available for Phase 2 HDQPSK modulation type).

`HDQPSK_P25` sets the filter to HDQPSK-P25 (only available for Phase 2 HDQPSK modulation type)

`C4FM_P25` sets the filter to C4FM-P25 (only available for Phase 1 C4FM modulation type)

Returns

`OFF` means the filter is set to None.

`USER1` means the filter is set to User Meas Filter 1.

`USER2` means the filter is set to User Meas Filter 2.

`USER3` means the filter is set to User Meas Filter 3.

`UOTH` means the filter is set to User other.

`RECT` means the filter is set to Rectangular (only available for Phase 2 HDQPSK modulation type).

`RRC` means the filter is set to Root Raised Cosine (only available for Phase 2 HDQPSK modulation type).

RCOS means the filter is set to Raised Cosine (only available for Phase 2 HDQPSK modulation type).

GAUS means the filter is set to Gaussian (only available for Phase 2 HDQPSK modulation type).

HDQPSK_P25 means the filter is set to HDQPSK-P25 (only available for Phase 2 HDQPSK modulation type)

C4FM_P25 means the filter is set to C4FM-P25 (only available for Phase 1 C4FM modulation type)

Examples

SENSe:P25:FILTER:MEASUREMENT? might return USER1, indicating that the Measurement Filter is set to User Meas Filter 1.

[SENSe]:P25:FILTer:REFerence

Sets or queries the P25 measurements Reference Filter setting. The User Meas Filter selections are predefined user filters.

Conditions

Measurement view: Any P25 measurement display

Group

Sense commands

Syntax

```
[SENSe]:P25:FILTer:REFerence{ OFF | USER1 | USER2 | USER3 | UOTH | RECT | RRC | RCOS | GAUS |
   HDQPSK_P25| C4FM_P25}
[SENSe]:P25:FILTer:REFerence?
```

Arguments

OFF sets the filter to None.

USER1 sets the filter to User Meas Filter 1.

USER2 sets the filter to User Meas Filter 2.

USER3 sets the filter to User Meas Filter 3.

UOTH sets the filter to User other.

RECT sets the filter to Rectangular (only available for Phase 2 HDQPSK modulation type).

RRC sets the filter to Root Raised Cosine (only available for Phase 2 HDQPSK modulation type).

RCOS sets the filter to Raised Cosine (only available for Phase 2 HDQPSK modulation type).

GAUS sets the filter to Gaussian (only available for Phase 2 HDQPSK modulation type).

HDQPSK_P25 sets the filter to HDQPSK-P25 (only available for Phase 2 HDQPSK modulation type)

C4FM_P25 sets the filter to C4FM-P25 (only available for Phase 1 C4FM modulation type)

Returns

OFF means the filter is set to None.

USER1 means the filter is set to User Meas Filter 1.

USER2 means the filter is set to User Meas Filter 2.

USER3 means the filter is set to User Meas Filter 3. UOTH means the filter is set to User other.

RECT means the filter is set to Rectangular (only available for Phase 2 HDQPSK modulation type).

RCOS means the filter is set to Raised Cosine (only available for Phase 2 HDQPSK modulation type).

GAUS means the filter is set to Gaussian (only available for Phase 2 HDQPSK modulation type).

Examples

SENSe:P25:FILTER:REFERENCE? might return USER1, indicating that the Reference Filter is set to User Meas Filter 1.

[SENSe]:P25:FREQuency:ERRor

Sets or queries the P25 Frequency Offset (when Auto is off).

Conditions

Measurement view: Any P25 measurement display Frequency Offset set to manual (Auto off).

Group

Sense commands

Syntax

```
[SENSe]:P25:FREQuency:ERRor <NR2>
[SENSe]:P25:FREQuency:ERRor?
```

Related Commands

Arguments

<NR2> = a floating point value without an exponent, which is the fixed frequency offset in Hz.

Returns

<NR2> = a floating point value without an exponent, which is the fixed frequency offset in Hz.

Examples

SENSe:P25:FREQUENCY:ERROR? might return 749.9999390000, indicating that the frequency offset is 750.0 Hz.

SENSe:P25:FREQUENCY:ERROR 700.99 sets the frequency offset the 701.0 Hz.

[SENSe]:P25:FREQuency:ERRor:AUTO

Sets or queries if the P25 Frequency Error is set to Auto.

Conditions

Measurement view: Any P25 measurement display

Group

Sense commands

Syntax

```
[SENSe]:P25:FREQuency:ERRor:AUTO { 0 | 1 | OFF | ON }
[SENSe]:P25:FREQuency:ERRor:AUTO?
```

Arguments

OFF or 0 means the Auto is off.

ON or 1 means the Auto is on.

Returns

0 means that Auto is off.

1 means Auto is on.

Examples

`SENSe:P25:FREQUENCY:ERROR:AUTO?` might return `1`, indicating that the Frequency Error is set to Auto.

`SENSe:P25:FREQUENCY:ERROR:AUTO 0` turns off Auto setting for Frequency Error.

[SENSe]:P25:MODUlation:CHANnel:([BANDwidth]|:BWIDth)]? (Query Only)

Queries the actual P25 Preset channel bandwidth.

Conditions

Measurement view: Any P25 measurement display

Group

Sense commands

Syntax

```
[SENSe]:P25:MODUlation:CHANnel:([BANDwidth]|:BWIDth)]?
```

Returns

Bandwidth in kHz, using "_" to denote “.”. For example, 12_5 means 12.5 kHz.

Examples

`SENSe:P25:MODU:CHAN:BAND?` will return `12_5`, indicating that the channel bandwidth is 12.5 kHz.

[SENSe]:P25:MODUlation:STANdard

Sets or queries the P25 modulation standard.

Conditions

Measurement view: Any P25 measurement display

Group

Sense commands

Syntax

```
[SENSe]:P25:MODUlation:STANdard{PHASE1|PHASE2}
[SENSe]:P25:MODUlation:STANdard?
```

Related Commands

SENSe:P25:MODUlation:TYPE (on page 2231)

Arguments

`PHASE1` is the Phase 1 standard.

`PHASE2` is the Phase 2 standard.

Returns

PHASE1 is the Phase 1 standard.

PHASE2 is the Phase 2 standard.

Examples

`SENSe:P25:MODULATION:STANDARD?` might return `PHASE2`, indicating that the P25 modulation standard is set to Phase 2.

SENSe:P25:MODUlation:TYPE

Sets or queries the P25 modulation type.

Conditions

Measurement view: Any P25 measurement display

Group

Sense commands

Syntax

```
SENSe:P25:MODUlation:TYPE {C4FM|HCPM|HDQPSK}
SENSe:P25:MODUlation:TYPE?
```

Related Commands

[SENSe]:P25:MODUlation:STANdard (on page 2230)

Arguments

`C4FM`: this is a Phase 1 modulation standard type. Modulation Phase must be set to PHASE1.

`HCPM`: this is a Phase 2 modulation standard type. Modulation Phase must be set to PHASE2.

`HDQPSK`: this is a Phase 2 modulation standard type. Modulation Phase must be set to PHASE2.

Returns

C4FM: this is a Phase 1 modulation standard type.

HCPM: this is a Phase 2 modulation standard type.

HDQPSK: this is a Phase 2 modulation standard type.

Examples

`SENSe:P25:MODULATION:TYPE?` might return `HDQPSK`, indicating that the P25 modulation type is set to HDQPSK, which is a Phase 2 modulation standard.

[SENSe]:P25:SRATe

Sets or queries the Symbol Rate value for the P25 measurements.

Conditions

Measurement view: Any P25 measurement display

Group

Sense commands

Syntax

```
[SENSe]:P25:SRATe <value>
[SENSe]:P25:SRATe?
```

Arguments

<value> is the symbol rate value in kHz.

Returns

<NR3> = a floating point value with an exponent, which is the symbol rate value in kHz.

Examples

SENSe:P25:SRATE? might return 6.0000000000E+3, indicating that the symbol rate is 6 kHz.

[SENSe]:P25:SUMMary:CLEar:RESults (No Query Form)

Clears the results of the P25 Summary display.

Conditions

Measurement view: P25 Summary

Group

Sense commands

Syntax

```
[SENSe]:P25:SUMMary:CLEar:RESults
```

Arguments

None

Examples

SENSe:P25:SUMMARY:CLEAR:RESULTS clears the results in the P25 Summary display.

[SENSe]:P25:SUMMary:LIMIt:FA

Sets or queries for the On/Off state of limit comparison and the upper limit value for Operating Frequency Accuracy. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions

Measurement view: P25 Summary

Group

Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FA <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FA?
```

Arguments

<NR1> = 0 to deselect the measurement and 1 to select the measurement.

<NR2> = 0 to 4

Returns

<NR1>,<NR2>

Examples

SENSe:P25:SUMMARY:LIMIT:FA? might return "1,2" <1>, which indicates measurement is enabled (turned on) for limit comparison, and the upper limit for Operating Frequency Accuracy is 2.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:AVERage:TERRor:ONE

Sets or queries for the On/Off state of limit comparison and the upper limit value for Average t_error_1. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions

Measurement view: P25 Summary

Group

Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:AVERage:TERRor:ONE <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:AVERage:TERRor:ONE?
```

Arguments

<NR1> = 0 to deselect the measurement and 1 to select the measurement.

<NR2> = 0 to 25

Returns

<NR1>,<NR3>

Examples

SENSe:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:AVERAGE:TERROR:ONE? might return "0,25.0000000000E-6",<0> indicating measurement is disabled (turned off) for limit comparison, and the upper limit for Average t_error_1 is <25.0000000000E-6>.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:AVERage:TERRor:ZERO

Sets or queries for the On/Off state of limit comparison and the upper limit value for Average t_error_0. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions

Measurement view: P25 Summary

Group

Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:AVERage:TERRor:ZERO <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:AVERage:TERRor:ZERO?
```

Arguments

<NR1> = 0 to deselect the measurement and 1 to select the measurement.

<NR2> = 0 to 25

Returns

<NR1>,<NR3>

Examples

SENSe:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:AVERAGE:TERROR:ZERO? might return "0,25.0000000000E-6",<0> indicating measurement is disabled (turned off) for limit comparison, and the upper limit for Average t_error_0 is <25.0000000000E-6>.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:LOGIc:CHANnel:OFF:SLOT

Sets or queries for the On/Off state of limit comparison and the upper limit value for HCPM Transmitter Logic channel offslot. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions

Measurement view: P25 Summary

Group

Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:LOGIc:CHANnel:OFF:SLOT <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:LOGIc:CHANnel:OFF:SLOT?
```

Arguments

<NR1> = 0 to deselect the measurement and 1 to select the measurement.

<NR2> = 0 to –57

Returns

<NR1>,<NR2>

Examples

SENSe:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:LOGIC:CHANNEL:OFF:SLOT? might return "0,–57",<0>, which indicates measurement is disabled (turned off) for limit comparison, and the upper limit for HCPM Transmitter Logic channel offslot is <-57>.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:LOGIc:CHANnel:PEAK:ACPR

Sets or queries for the On/Off state of limit comparison and the upper limit value for HCPM Transmitter Logic channel Peak ACPR. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions

Measurement view: P25 Summary

Group

Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:LOGIc:CHANnel:PEAK:ACPR <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:LOGIc:CHANnel:PEAK:ACPR?
```

Arguments

<NR1> = 0 to deselect the measurement and 1 to select the measurement.

<NR2> = 0 to 35

Returns

<NR1>,<NR2>

Examples

SENSe:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:LOGIC:CHANNEL:PEAK:ACPR? might return "0,35",<0> indicating measurement is disabled (turned off) for limit comparison, and the upper limit for HCPM Transmitter Logic channel Peak ACPR is <35>.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PEAK:NEGAtive:LO

Sets or queries for the On/Off state of limit comparison and the lower limit value for Frequency Deviation, negative peak. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions

Measurement view: P25 Summary

Group

Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PEAK:NEGAtive:LO <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PEAK:NEGAtive:LO?
```

Arguments

<NR1> = 0 to deselect the measurement and 1 to select the measurement.

<NR2> = 0 to 2544

Returns

<NR1>,<NR2>

Examples

SENSe:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:PEAK:NEGATIVE:LO? might return "1,2544",<1> indicates measurement is enabled (turned on) for limit comparison, and the lower limit for Frequency Deviation, negative peak, is <2544>.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PEAK:NEGAtive:UP

Sets or queries for the On/Off state of limit comparison and the upper limit value for Frequency Deviation, negative peak. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions

Measurement view: P25 Summary

Group

Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PEAK:NEGAtive: UP <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PEAK:NEGAtive: UP?
```

Arguments

<NR1> = 0 to deselect the measurement and 1 to select the measurement.

<NR2> = 0 to 3111

Returns

<NR1>,<NR2>

Examples

SENSe:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:PEAK:NEGATIVE:UP? might return "1,3111", which indicates measurement is enabled (turned on) for limit comparison, and the upper limit for Frequency Deviation, negative peak, is <3111>.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PEAK:POSItive:LO

Sets or queries for the On/Off state of limit comparison and the lower limit value for Frequency Deviation, positive peak. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions

Measurement view: P25 Summary

Group

Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PEAK:POSItive:LO <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PEAK:POSItive:LO?
```

Arguments

<NR1> = 0 to deselect the measurement and 1 to select the measurement.

<NR2> = 0 to 2544

Returns

<NR1>,<NR2>

Examples

SENSe:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:PEAK:POSITIVE:LO? might return "1,2544", which indicates measurement is enabled (turned on) for limit comparison and the lower limit for Frequency Deviation, positive peak, is <2544>.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PEAK:POSItive:UP

Sets or queries for the On/Off state of limit comparison and the upper limit value for Frequency Deviation, positive peak. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions

Measurement view: P25 Summary

Group

Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PEAK:POSItive:UP <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PEAK:POSItive:UP?
```

Arguments

<NR1> = 0 to deselect the measurement and 1 to select the measurement.

<NR2> = 0 to 3111 Hz

Returns

<NR1>,<NR2>

Examples

SENSe:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:PEAK:POSITIVE:UP? might return "1,3111" which indicates measurement is enabled (turned on) for limit comparison, and the upper limit for Frequency Deviation, positive peak, is <3111>.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PHASe[1/2]:F[1/2/3/4]:LO

Sets or queries for the On/Off state of limit comparison and the lower limit value for Frequency Deviation, f1/f2/f3/f4 for Phase1 or Phase2. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions

Measurement view: P25 Summary

Group

Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PHASe[1/2]:F[1/2/3/4]:LO <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PHASe[1/2]:F[1/2/3/4]:LO?
```

Arguments

<NR1> =0 to deselect the measurement and 1 to select the measurement.

<NR2> =0 to 2544

Returns

<NR1>,<NR2>

Examples

SENSe:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:PHASE[1/2]:F[1/2/3/4]:LO? might return "1,2544", indicating measurement is enabled (turned on) for limit comparison, and the upper limit for Frequency Deviation is <2544>.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PHASe[1/2]:F[1/2/3/4]:LO

Sets or queries for two values namely, the On/Off state of limit comparison and the lower limit value for Frequency deviation, f1/f2/f3/f4 for Phase1 or Phase2. The limit comparison can be enabled (turned On) or disabled (turned off).

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PHASe[1/2]: F[1/2/3/4]:LO <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PHASe[1/2]: F[1/2/3/4]:LO?
```

Arguments

<Integer, Integer>

Examples

SENSe:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:PHASE[1/2]:F[1/2/3/4]:LO? returns "1,2544", Indicating the measurement is enabled for limit comparison and the lower limit for Frequency deviation is <2544>.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PHASe[1/2]:F[1/2/3/4]:UP

Sets or queries for the On/Off state of limit comparison and the upper limit value for Frequency Deviation, f1/f2/f3/f4 for Phase1 or Phase2. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions

Measurement view: P25 Summary

Group

Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PHASe[1/2]:F[1/2/3/4]:UP <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PHASe[1/2]:F[1/2/3/4]:UP?
```

Arguments

<NR1> =0 to deselect the measurement and 1 to select the measurement.

<NR2> =0 to 3111

Returns

<NR1>,<NR2>

Examples

SENSe:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:PHASE[1/2]:F[1/2/3/4]:UP? might return "1,3111", indicating measurement is enabled (turned on) for limit comparison, and the upper limit for Frequency Deviation is 3111.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PHASe[1/2]:F[1/2/3/4]:UP

Sets or queries for two values namely, the On/Off state of limit comparison and the upper limit value for Frequency deviation, f1/f2/f3/f4 for Phase1 or Phase2. The limit comparison can be enabled (turned On) or disabled (turned off).

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PHASe[1/2]: F[1/2/3/4]:UP <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PHASe[1/2]:F[1/2/3/4]:UP?
```

Arguments

<Integer, Integer>

Examples

SENSe:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:PHASE[1/2]:F[1/2/3/4]:UP? returns “1,3111 “ Indicating the measurement is enabled for limit comparison and the upper limit for Frequency deviation is <3111>.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMAX:OFF

Sets or queries for the On/Off state of limit comparison and the upper limit value for Pmax-off. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions

Measurement view: P25 Summary

Group

Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMAX:OFF <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMAX:OFF?
```

Arguments

<NR1> =0 to deselect the measurement and 1 to select the measurement.

<NR2> =0 to 4

Returns

<NR1>,<NR2>

Examples

SENSe:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:PMAX:OFF? might return "0,4" <0>, indicating measurement is disabled (turned off) for limit comparison, and the upper limit for Pmax-off is <4>.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMAX:OFF

Sets or queries for two values namely, the On/Off state of limit comparison and the upper limit value for Pmax-off. The limit comparison can be enabled (turned On) or disabled (turned off).

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMAX:OFF <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMAX:OFF?
```

Arguments

<Integer, Integer>

Examples

SENSe:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:PMAX:OFF? returns "0,4", which indicates Measurement is disabled (turned Off) for limit comparison, and the upper limit for Pmax-off is <4>.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMAX:ON

Sets or queries for the On/Off state of limit comparison and the upper limit value for Pmax-on. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions

Measurement view: P25 Summary

Group

Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMAX:ON <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMAX:ON?
```

Arguments

<NR1> =0 to deselect the measurement and 1 to select the measurement.

<NR2> =0 to 4

Returns

<NR1>,<NR2>

Examples

SENSe:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:PMAX:ON? might return "0,4", indicating measurement is disabled (turned off) for limit comparison, and the upper limit for Pmax-on is 4.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMAX:ON

Sets or queries for two values namely, the On/Off state of limit comparison and the upper limit value for Pmax-on. The limit comparison can be enabled (turned On) or disabled (turned off).

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMAX:ON<NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMAX:ON?
```

Arguments

<Integer, Integer>

Examples

SENSe:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:PMAX:ON? returns "0,4", which indicates Measurement is disabled (turned Off) for limit comparison, and the upper limit for Pmax-on is 4.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMAX:SS

Sets or queries for the On/Off state of limit comparison and the upper limit value for Pss-max. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions

Measurement view: P25 Summary

Group

Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMAX:SS <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMAX:SS?
```

Arguments

<NR1> = 0 to deselect the measurement and 1 to select the measurement.

<NR2> = 0 to 1

Returns

<NR1>,<NR2>

Examples

SENSe:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:PMAX:SS? might return "0,1", indicating measurement is disabled (turned off) for limit comparison, and the upper limit for Pss-max is 1.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMIN:SS

Sets or queries for the On/Off state of limit comparison and the upper limit value for Pss-min. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions

Measurement view: P25 Summary

Group

Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMIN:SS <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMIN:SS?
```

Arguments

<NR1> = 0 to deselect the measurement and 1 to select the measurement.

<NR2> = 0 to -3

Returns

<NR1>,<NR2>

Examples

SENSe:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:PMIN:SS? might return "0,-3", indicating measurement is disabled (turned off) for limit comparison, and the upper limit for Pss-min is -3.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter:AVERage:ATTK:ENCOder

Sets or queries for the On/Off state of limit comparison and the upper limit value for Transmitter Average Attack encoder. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions

Measurement view: P25 Summary

Group

Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter: AVERage:ATTK:ENCOder <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter: AVERage:ATTK:ENCOder?
```

Arguments

<NR1> = 0 to deselect the measurement and 1 to select the measurement.

<NR2> = 0 to 100

Returns

<NR1>,<NR2>

Examples

SENSe:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:TRANSMITTER:AVERAGE:ATTK:ENCODER? might return "0,100", indicating measurement is disabled (turned off) for limit comparison, and the upper limit for Transmitter Average Attack encoder is 100.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter:AVERage:ATTK:ENCOder:BI

Sets or queries for the On/Off state of limit comparison and the upper limit value for Transmitter Average Attack encoder, busy/idle. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions

Measurement view: P25 Summary

Group

Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter: AVERage:ATTK:ENCOder:BI <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter: AVERage:ATTK:ENCOder:BI?
```

Arguments

<NR1> = 0 to deselect the measurement and 1 to select the measurement.

<NR2> = 0 to 30

Returns

<NR1>,<NR2>

Examples

SENSe:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:TRANSMITTER:AVERAGE:ATTK:ENCODER:BI? might return "0,30", which indicates measurement is disabled (turned off) for limit comparison and the upper limit for Transmitter Average Attack encoder, busy/idle, is 30.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter:AVERage:ATTK:POWEr

Sets or queries for the On/Off state of limit comparison and the upper limit value for Transmitter Average Attack power. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions

Measurement view: P25 Summary

Group

Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter: AVERage:ATTK:POWEr <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter: AVERage:ATTK:POWEr?
```

Arguments

<NR1> = 0 to deselect the measurement and 1 to select the measurement.

<NR2> = 0 to 50

Returns

<NR1>,<NR2>

Examples

SENSe:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:TRANSMITTER:AVERAGE:ATTK:POWER? might return "0,50", which indicates measurement is disabled (turned off) for limit comparison and the upper limit for Transmitter Average Attack power is 50.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter:AVERage:ATTK:POWEr:BI

Sets or queries for the On/Off state of limit comparison and the upper limit value for Transmitter Average Attack power, busy/idle. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions

Measurement view: P25 Summary

Group

Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter: AVERage:ATTK:POWEr:BI <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter: AVERage:ATTK:POWEr:BI?
```

Arguments

<NR1> = 0 to deselect the measurement and 1 to select the measurement.

<NR2> = 0 to 30

Returns

<NR1>,<NR2>

Examples

SENSe:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:TRANSMITTER:AVERAGE:ATTK:POWER:BI? might return "0,30", which indicates measurement is disabled (turned off) for limit comparison, and the upper limit for Transmitter Average Attack power, busy/idle, is 30.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter:AVERage:THRUput:DLAY

Sets or queries for the On/Off state of limit comparison and the upper limit value for Transmitter Average throughput delay. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions

Measurement view: P25 Summary

Group

Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter: AVERage:THRUput:DLAY <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter: AVERage:THRUput:DLAY?
```

Arguments

<NR1> = 0 to deselect the measurement and 1 to select the measurement.

<NR2> = 0 to 125

Returns

<NR1>,<NR2>

Examples

SENSe:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:TRANSMITTER:AVERAGE:THRUPUT:DLAY? might return "0,125", which indicates measurement is disabled (turned off) for limit comparison, and the upper limit for Transmitter Average throughput delay is 125.

[SENSe]:P25:SUMMary:LIMIt:MF

Sets or queries for the On/Off state of limit comparison and the upper limit value for Modulation Fidelity. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions

Measurement view: P25 Summary

Group

Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:MF <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:MF?
```

Arguments

<NR1> = 0 to deselect the measurement and 1 to select the measurement.

<NR2> = 0 to 4

Returns

<NR1>,<NR2>

Examples

SENSe:P25:SUMMARY:LIMIT:MF? might return "1,4", which indicates measurement is enabled (turned on) for limit comparison, and the upper limit for Modulation Fidelity is 4.

[SENSe]:P25:SUMMary:LIMIt:RF:OUTPower

Sets or queries for the On/Off state of limit comparison and the upper limit value for RF Output Power. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions

Measurement view: P25 Summary

Group

Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:RF:OUTPower <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:RF:OUTPower?
```

Arguments

<NR1> = 0 to deselect the measurement and 1 to select the measurement.

<NR2> = 0 to 3

Returns

<NR1>,<NR2>

Examples

SENSe:P25:SUMMARY:LIMIT:RF:OUTPOWER? might return "1,3", which indicates measurement is enabled (turned on) for limit comparison, and the upper limit for RF Output Power is 3.

[SENSe]:P25:SUMMary:LIMIt:SRA

Sets or queries for the On/Off state of limit comparison and the upper limit value for Symbol Rate Accuracy. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions

Measurement view: P25 Summary

Group

Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:SRA <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:SRA?
```

Arguments

<NR1> = 0 to deselect the measurement and 1 to select the measurement.

<NR2> = 0 to 10

Returns

<NR1>,<NR2>

Examples

SENSe:P25:SUMMARY:LIMIT:SRA? might return "1,10", which indicates measurement is enabled (turned on) for limit comparison and the upper limit for Symbol Rate Accuracy is 10.

[SENSe]:P25:SYMBols:POINts

Sets or queries the Points/Symbols to display in the P25 measurement view.

Conditions

Measurement view: Any P25 measurement display

Group

Sense commands

Syntax

```
[SENSe]:P25:SYMBols:POINts{ONE|TWO|FOUR|EIGHt}
[SENSe]:P25:SYMBols:POINts?
```

Arguments

ONE set the Points/Symbol trace setting to 1.

TWO set the Points/Symbol trace setting to 2.

FOUR set the Points/Symbol trace setting to 4.

EIGHt set the Points/Symbol trace setting to 8.

Returns

ONE means the Points/Symbol trace setting is set to 1.

TWO means the Points/Symbol trace setting is set to 2.

FOUR means the Points/Symbol trace setting to is set 4.

EIGH means the Points/Symbol trace setting to is set 8.

Examples

SENSe:P25:SYMBOLS:POINTS? might return FOUR, indicating that the Points/Symbol value is set to 4.

[SENSe]:P25:TEST:PATTern

Sets or returns the test pattern type in the P25 displays.

Conditions

Measurement views: Any Bluetooth display

Group

Sense commands

Syntax

```
[SENSe]:P25:TEST:PATTern{LDEViation|HDEViation|TPOTHER}
[SENSe]:P25:TEST:PATTern?
```

Arguments

`LDEViation` indicates the low deviation test pattern.

`HDEViation` indicates the high deviation test pattern.

`TPOTHER` indicates a user defined test pattern.

Examples

`SENSe:P25:TEST:PATTERN?` might return `LDEV`, indicating the selected test pattern type is the low deviation type.

[SENSe]:P25:TIMe:UNITs

Sets or queries the fundamental unit of time for the P25 measurements.

Conditions

Measurement view: Any P25 measurement display

Group

Sense commands

Syntax

```
[SENSe]:P25:TIMe:UNITs{SEC|SYMB}
[SENSe]:P25:TIMe:UNITs?
```

Arguments

`SEConds` sets the unit of time to seconds.

`SYMBols` sets the unit of time to symbols.

Returns

`SEC` means that the unit if time is set to seconds.

`SYMB` means that the unit if time is set to symbols.

Examples

`SENSe:P25:TIME:UNITS?` might return `SEC`, indicating that the unit of time is set to seconds.

[SENSe]:P25:TRIGgger:MEASurement

Sets or queries the Trigger Measurement from the P25 displays.

Conditions

Measurement views: P25 displays

Group

Sense commands

Syntax

```
[SENSe]:P25:TRIGgger:MEASurement <arg1>
[SENSe]:P25:TRIGgger:MEASurement?
```

Arguments

TATOBSYNC is the Time Alignment (tOB_sync measurement).

TAERR0 is the Time Alignment (t_error_0_calculation).

TAERR1 is the Time Alignment (t_error_1_calculation).

TIMEASNONE is None.

Returns

See Arguments

Examples

SENSe:P25:TRIGGGER:MEASUREMENT TATOBSYNC selects the Time Alignment (tOB_sync measurement) option when Modulation Params Standard is Phase2 and Modulation Type is C4FM.

[SENSe]:P25:TRIGgger:MEASurement:TOBSync

Sets or returns the tOB_sync (measured) value in the P25 displays.

Conditions

Measurement views: P25 displays

Group

Sense commands

Syntax

```
[SENSe]:P25:TRIGgger:MEASurement:TOBSync <value>
```

Arguments

None

Examples

SENSe:P25:TRIGGGER:MEASUREMENT:TOBSYNC? might return 6.00000000, which is the tOB_sync(measured) value.

[SENSe]:P25:TRIGgger:MEASurement:TOBSync:AUTO

Sets or queries the Auto checkbox selection from the P25 displays.

Conditions

Measurement views: P25 displays

Group

Sense commands

Syntax

```
[SENSe]:P25:TRIGgger:MEASurement:TOBSync:AUTO <value>
```

Arguments

0 clears the checkbox.

1 checks the checkbox.

Returns

0 or 1, **<NRf>**

Examples

`SENSe:P25:TRIGGGER:MEASUREMENT:TOBSYNC:AUTO 1` enables the Auto checkbox in Trig Meas tab.

[SENSe]:P25:TRIGgger:MEASurement:TRANsient:FREQuency:BEHAviour

Sets or queries the Trigger Measurements option Phase1-Transient Frequency Behavior from the P25 displays.

Conditions

Measurement views: P25 displays

Group

Sense commands

Syntax

`[SENSe]:P25:TRIGgger:MEASurement:TRANsient:FREQuency:BEHAviour {RISE|FALL}`

Arguments

`RISE` sets the Transient Frequency Behavior to Off to On.

`FALL` sets the Transient Frequency Behavior to On to Off.

Returns

RISE means the Transient Frequency Behavior is Off to On.

FALL means the Transient Frequency Behavior is On to Off.

Examples

`SENSe:P25:TRIGGGER:MEASUREMENT:TRANSIENT:FREQUENCY:BEHAVIOUR?` might return `RISE` if the Transient Frequency behavior Off to On option is chosen in the Transient Frequency.

[SENSe]:P25:TRIGgger:MEASurement:TRANsient:FREQuency:BEHAviour:TIME:OFF

Sets or queries the ‘t off’ value when the Transient Frequency option is Transient Frequency Behavior On to Off for Phase1-Transient Frequency Behavior from the P25 displays.

Conditions

Measurement views: P25 displays

Group

Sense commands

Syntax

`[SENSe]:P25:TRIGgger:MEASurement:TRANsient:FREQuency: BEHAviour:TIME:OFF <arg1>`
`[SENSe]:P25:TRIGgger:MEASurement:TRANsient:FREQuency:BEHAviour:TIME:OFF?`

Arguments

`<arg1>`

Returns

<arg1>::<NRf>

Examples

SENSe:P25:TRIGGGER:MEASUREMENT:TRANSIENT:FREQUENCY:BEHAVIOUR:TIME:OFF 2 sets the "t off" value to 2.

[SENSe]:P25:TRIGgger:MEASurement:TRANsient:FREQuency:BEHAviour:TIME:OFF:AUTO

Sets or queries the Auto checkbox selection when Transient Frequency is chosen as Transient Frequency Behavior On to Off from the P25 displays.

Conditions

Measurement views: P25 displays

Group

Sense commands

Syntax

[SENSe]:P25:TRIGgger:MEASurement:TRANsient:FREQuency: BEHAviour:TIME:OFF:AUTO <value>

Arguments

0 unchecks the checkbox.

1 checks the checkbox.

Returns

0 or 1, <NRf>

Examples

SENSe:P25:TRIGGGER:MEASUREMENT:TRANSIENT:FREQUENCY:BEHAVIOUR:TIME:OFF:AUTO 1 enables the Auto checkbox in Trig measurement tab.

[SENSe]:P25:TRIGgger:MEASurement:TRANsient:FREQuency:BEHAviour:TIME:ON

Sets or queries the ‘t on’ value when the Transient Frequency option is Transient Frequency Behavior Off to On for Phase1-Transient Frequency Behavior from the P25 displays.

Conditions

Measurement views: P25 displays

Group

Sense commands

Syntax

[SENSe]:P25:TRIGgger:MEASurement:TRANsient:FREQuency:BEHAviour:TIME:ON <arg1>
[SENSe]:P25:TRIGgger:MEASurement:TRANsient:FREQuency:BEHAviour:TIME:ON?

Arguments

<arg1>

Examples

SENSe:P25:TRIGGGER:MEASUREMENT:TRANSIENT:FREQUENCY:BEHAVIOUR:TIME:ON 2 sets the "t on" value to 2.

[SENSe]:P25:TRIGgger:MEASurement:TRANsient:FREQuency:BEHAviour:TIME:ON:AUTO

Sets or queries the Auto checkbox selection when Transient Frequency is chosen as Transient Frequency behavior Off to On from the P25 displays.

Conditions

Measurement views: P25 displays

Group

Sense commands

Syntax

[SENSe]:P25:TRIGgger:MEASurement:TRANsient:FREQuency: BEHAviour:TIME:ON:AUTO <value>

Arguments

0 unchecks the checkbox.

1 checks the checkbox.

Returns

0 or 1, <NRf>

Examples

SENSe:P25:TRIGGGER:MEASUREMENT:TRANSIENT:FREQUENCY:BEHAVIOUR:TIME:ON:AUTO 1 enables the Auto checkbox in Trig measurement tab.

[SENSe]:PERRor[:MEASview<y>]:SELEcted:SOURce

Sets or queries the Source of Phase error versus Time for a selected Measurement view number.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Phase error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Phase error versus Time).

Group

Sense commands

Syntax

[SENSe]:PERRor[:MEASview<y>]:SELEcted:SOURce {"Source 1" | "Source 2" | "Source 3" | "Source 4"}
[SENSe]:PERRor[:MEASview<y>]:SELEcted:SOURce?

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1.

Examples

`SENSe:PERRor:MEASview1:SELEcted:SOURce?` queries the Source for Phase error versus Time Measurement View 1.

[SENSe]:PHVTime[:MEASview<y>]:CLEar:RESults (No Query Form)

Restarts multi-trace functions (Average and Max/Min Hold).

Conditions

Measurement views: Phase versus Time

Group

Sense commands

Syntax

`[SENSe]:PHVTime[:MEASview<y>]:CLEar:RESults`

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Examples

`SENSe:PHVTIME:MEASview1:CLEAR:RESULTS` restarts multi-trace functions.

[SENSe]:PHVTime[:MEASview<y>]:COLor

Sets or queries the color palette of three-dimensional graphs.

Conditions

Measurement views: Phase versus Time.

Group

Sense commands

Syntax

```
[SENSe]:PHVTime[:MEASview<y>]:COLor {RED | GREen | BLUe | CYAN | BCYan | YELLow | MAGenta | GRAY | TEMPerature | SPECtral}
```

`[SENSe]:PHVTime[:MEASview<y>]:COLor?`

Arguments

The following table lists the arguments.

Argument	Description
`RED`	Red

Argument	Description
GREen	Green
BLUe	Blue
CYAN	Cyan
BCYan	Binary cyan
YELLow	Yellow
MAGenta	Magenta
GRAY	Gray
TEMPerature	Temperature
SPECtral	Spectral

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`SENSE:PHVTIME:COLOR TEMPERATURE` selects the temperature color palette.

[SENSe]:PHVTime[:MEASview<y>]:COLor:MAXimum

Sets or queries the maximum value of the color axis in the Phase versus Time measurement.

Conditions

Measurement views: Phase versus Time.

Group

Sense commands

Syntax

`[SENSe]:PHVTime[:MEASview<y>]:COLor:MAXimum <value>`

`[SENSe]:PHVTime[:MEASview<y>]:COLor:MAXimum?`

Arguments

`<value>::=<NRf>` specifies the maximum value of the color axis.

Range: The minimum value to 100%.

The minimum value is set using the `[SENSe]:PHVTime[:MEASview<y>]:COLor:MINimum` command.

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`SENSE:PHVTIME:COLOR:MAXIMUM 90` sets the maximum value of the color axis to 90%.

[SENSe]:PHVTime[:MEASview<y>]:COLor:MINimum

Sets or queries the minimum value of the color axis in the Phase versus Time measurement.

Conditions

Measurement views: Phase versus Time.

Group

Sense commands

Syntax

`[SENSe]:PHVTime[:MEASview<y>]:COLor:MINimum <value>`

`[SENSe]:PHVTime[:MEASview<y>]:COLor:MINimum?`

Arguments

`<value>::=<NRf>` specifies the minimum value of the color axis.

Range: 0% to the maximum value.

The maximum value is set using the `[SENSe]:PHVTime[:MEASview<y>]:COLor:MAXimum` command.

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`SENSE:PHVTIME:COLOR:MINIMUM 10` sets the minimum value of the color axis to 10%.

[SENSe]:PHVTime[:MEASview<y>]:COLor:REVerse[:STATe]

Determines whether to reverse the color palette in the Phase versus Time view. When reversed, the highest density takes the color normally used for the lowest.

Conditions

Measurement views: Phase versus Time.

Group

Sense commands

Syntax

`[SENSe]:PHVTime[:MEASview<y>]:COLor:REVerse[:STATe] {OFF | ON | 0 | 1}`

`[SENSe]:PHVTime[:MEASview<y>]:COLor:REVerse[:STATe]?`

Arguments

`OFF` or `0` uses the color palette in its normal order.

`ON` or `1` reverses the color palette.

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`SENSE:PHVTIME:COLOR:REVERSE:STATE ON` reverses the color palette.

[SENSe]:PHVTime[:MEASview<y>]:FREQuency:CENTer

Sets or queries the center frequency in the Phase versus Time measurement.

NOTE: The center, start and stop frequencies are set interlocking each other with the following relationships: (start frequency) = (center frequency) - (span)/2 and (stop frequency) = (center frequency) + (span)/2.

Conditions

Measurement views: Phase versus Time

Group

Sense commands

Syntax

```
[SENSe]:PHVTime[:MEASview<y>]:FREQuency:CENTer <value>
[SENSe]:PHVTime[:MEASview<y>]:FREQuency:CENTer?
```

Related Commands

[SENSe]:PHVTime[:MEASview<y>]:FREQuency:STARt (on page 2257)

[SENSe]:PHVTime:MEASview<y>:FREQuency:STOP (on page 2259)

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

`<value>::<NRf>` specifies the center frequency.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples

`SENSe:PHVTIME:MEASview1:FREQUENCY:CENTER 7.5GHz` sets the center frequency to 7.5 GHz.

[SENSe]:PHVTime[:MEASview<y>]:FREQuency:SPAN

Sets or queries the frequency span in the Phase versus Time measurement.

Conditions

Measurement views: Phase versus Time

Group

Sense commands

Syntax

```
[SENSe]:PHVTime[:MEASview<y>]:FREQuency:SPAN <value>
[SENSe]:PHVTime[:MEASview<y>]:FREQuency:SPAN?
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<value>::<NRf> is the frequency span.

Range: 10 Hz to 40 MHz (Standard) / 110 MHz (Option 110)

Examples

SENSe:PHVTIME:MEASview1:FREQUENCY:SPAN 20MHz sets the span to 20 MHz.

[SENSe]:PHVTime[:MEASview<y>]:FREQuency:SPAN:LINK

Sets or queries the frequency span link in the Phase versus Time measurement.

Conditions

Measurement views: Phase versus Time

Group

Sense commands

Syntax

```
[SENSe]:PHVTime[:MEASview<y>]:FREQuency:SPAN:LINK <value>
[SENSe]:PHVTime[:MEASview<y>]:FREQuency:SPAN:LINK?
```

Arguments

OFF or 0 specifies that Link to Span is not checked.

ON or 1 specifies that Link to Span is checked .

Examples

SENSe:PHVTime:MEASview2:FREQuency:SPAN:LINK? might return 0 or 1 based on condition of Link to Span checkbox.

[SENSe]:PHVTime[:MEASview<y>]:FREQuency:STARt

Sets or queries the measurement start frequency (left edge on the graph) in the Phase versus Time measurement.

The center, start and stop frequencies are set interlocking each other. Refer to the [SENSe]:PHVTime[:MEASview<y>]:FREQuency:CENTer command.

Conditions

Measurement views: Phase versus Time

Group

Sense commands

Syntax

```
[SENSe]:PHVTime[:MEASview<y>]:FREQuency:STARt <value>
[SENSe]:PHVTime[:MEASview<y>]:FREQuency:STARt?
```

Related Commands

[SENSe]:PHVTime[:MEASview<y>]:FREQuency:STOP (on page 2259)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<value>::<NRf> is the measurement start frequency.

Range: (center frequency) ± (span)/2.

Examples

SENSe:PHVTIME:MEASview1:FREQUENCY:START 6.95GHz sets the start frequency to 6.95 GHz.

[SENSe]:PHVTime[:MEASview<y>]:FREQuency:STEP

Sets or queries the frequency step size (the amount per press by which the up or down key changes the setting value). Programming a specified step size sets [SENSe]:PHVTime[:MEASview<y>]:FREQuency:STEP:AUTO OFF.

Conditions

Measurement views: Phase versus Time

Group

Sense commands

Syntax

```
[SENSe]:PHVTime[:MEASview<y>]:FREQuency:STEP <value>
[SENSe]:PHVTime[:MEASview<y>]:FREQuency:STEP?
```

Related Commands

[SENSe]:PHVTime[:MEASview<y>]:FREQuency:STEP:AUTO (on page 2258)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<value>::<NRf> specifies the frequency step size.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples

SENSe:PHVTIME:MEASview1:FREQUENCY:STEP 1.5kHz sets the step size to 1.5 kHz.

[SENSe]:PHVTime[:MEASview<y>]:FREQuency:STEP:AUTO

Determines whether to set the frequency step size automatically or manually.

Conditions

Measurement views: Phase versus Time

Group

Sense commands

Syntax

```
[SENSe]:PHVTime[:MEASview<y>]:FREQuency:STEP:AUTO { OFF | ON | 0 | 1 }
[SENSe]:PHVTime[:MEASview<y>]:FREQuency:STEP:AUTO?
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

OFF or 0 specifies that the frequency step size is set manually using the [SENSe]:PHVTime[:MEASview<y>]:FREQuency:STEP command.

ON or 1 specifies that the frequency step size is set automatically.

Examples

SENSe:PHVTIME:MEASview1:BANDWIDTH:AUTO ON sets the frequency step size automatically.

[SENSe]:PHVTime[:MEASview<y>]:FREQuency:STOP

Sets or queries the measurement stop frequency (right edge of the graph) in the Phase versus Time measurement.

The center, start and stop frequencies are set interlocking each other. Refer to the [SENSe]:PHVTime[:MEASview<y>]:FREQuency:CENTer command.

Conditions

Measurement views: Phase versus Time

Group

Sense commands

Syntax

```
[SENSe]:PHVTime[:MEASview<y>]:FREQuency:STOP <value>
[SENSe]:PHVTime[:MEASview<y>]:FREQuency:STOP?
```

Related Commands

[SENSe]:PHVTime[:MEASview<y>]:FREQuency:STARt (on page 2257)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<value>::<NRf> is the measurement stop frequency.

Range: (center frequency) ± (span)/2.

Examples

SENSe:PHVTIME:MEASview1:FREQUENCY:STOP 7.05GHz sets the stop frequency to 7.05 GHz.

[SENSe]:PHVTime[:MEASview<y>]:MAXTracepoints

Sets or queries the maximum trace points in the Phase versus Time measurement.

Conditions

Measurement views: Phase versus Time

Group

Sense commands

Syntax

```
[SENSe]:PHVTime[:MEASview<y>]:MAXTracepoints { ONEK | TENK | HUNDredk | NEVerdecimate }
[SENSe]:PHVTime[:MEASview<y>]:MAXTracepoints?
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

ONEK sets the maximum trace points to 1 k.

TENK sets the maximum trace points to 10 k.

HUNDredk sets the maximum trace points to 100 k.

NEVerdecimate never decimates the trace points.

Examples

SENSe:PHVTIME:MEASview1:MAXTRACEPOINTS TENK sets the maximum trace points to 10 k.

[SENSe]:PHVTime[:MEASview<y>]:OGRAm:COLor

Sets or queries the ogram color scheme.

Conditions

Applicable measurement view must be active.

Group

Sense commands

Syntax

```
[SENSe]:PHVTime[:MEASview<y>]:OGRAm:COLor {RED | GREen | BLUe | CYAN | BCYan | YELLow |
MAGenta | GRAY | TEMPerature | SPECtral}
[SENSe]:PHVTime[:MEASview<y>]:OGRAm:COLor?
```

Related Commands

[SENSe]:PHVTime[:MEASview<y>]:OGRAm:COLor:MAXimum (on page 2261)

Arguments

The following table lists the arguments.

Argument	Description
RED	Red
GREen	Green
BLUe	Blue
CYAN	Cyan

Argument	Description
BCYan	Binary cyan
YELLow	Yellow
MAGenta	Magenta
GRAY	Gray
TEMPerature	Temperature
SPECtral	Spectral

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Returns

{RED | GREen | BLUe | CYAN | BCYan | YELLow | MAGenta | GRAY | TEMPerature | SPECtral}

Examples

SENSe:PHVTime:MEASview1:OGRAm:COLor TEMPerature selects the temperature color palette.

[SENSe]:PHVTime[:MEASview<y>]:OGRAm:COLor:MAXimum

Sets or queries the maximum value of the color axis in the ogram.

Conditions

Applicable measurement view must be active.

Group

Sense commands

Syntax

[SENSe]:PHVTime[:MEASview<y>]:OGRAm:COLor:MAXimum <value>

[SENSe]:PHVTime[:MEASview<y>]:OGRAm:COLor:MAXimum?

Related Commands

[SENSe]:PHVTime[:MEASview<y>]:OGRAm:COLor:MINimum (on page 2262)

Arguments

<value>::=<NRf> specifies the maximum value of the color axis.

Range: -100 to +100 dBm.

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional. By default y is 1.

Returns

<NRf> specifies the maximum value of the color axis.

Examples

`SENSe:PHVTime:MEASview1:OGRAm:COLor:MAXimum 10` sets the maximum value of the color axis to 10 dBm.

[SENSe]:PHVTime[:MEASview<y>]:OGRAm:COLor:MINimum

Sets or queries the minimum value of the color axis in the ogram.

Conditions

Applicable measurement view must be active.

Group

Sense commands

Syntax

`[SENSe]:PHVTime[:MEASview<y>]:OGRAm:COLor:MINimum <value>`

`[SENSe]:PHVTime[:MEASview<y>]:OGRAm:COLor:MINimum?`

Related Commands

[SENSe]:PHVTime[:MEASview<y>]:OGRAm:COLor:MAXimum (on page 2261)

Arguments

`<value>::=<NRf>` specifies the minimum value of the color axis.

Range: -100 to +100 dBm.

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Returns

`<NRf>` specifies the minimum value of the color axis.

Examples

`SENSe:PHVTime:MEASview1:OGRAm:COLor:MINimum 10` sets the minimum value of the color axis to 10 dBm.

[SENSe]:PHVTime[:MEASview<y>]:OGRAm:COLor:REVerse

Sets or queries reverse color scheme state.

Conditions

Applicable measurement view must be active.

Group

Sense commands

Syntax

`[SENSe]:PHVTime[:MEASview<y>]:OGRAm:COLor:REVerse {OFF | ON | 0 | 1}`

`[SENSe]:PHVTime[:MEASview<y>]:OGRAm:COLor:REVerse?`

Related Commands

[SENSe]:PHVTime[:MEASview<y>]:OGRAm:COLor (on page 2260)

Arguments

`OFF` or `0` disables Reverse color scheme.

`ON` or `1` enables Reverse color scheme.

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional. By default y is 1.

Returns

`0` disables Reverse color scheme.

`1` enables Reverse color scheme.

Examples

`SENSe:PHVTime:MEASview1:OGRAm:COLor:REVerse OFF` disables reverse color scheme.

[SENSe]:PHVTime[:MEASview<y>]:PLOT

Sets or queries the Phase vs Time display mode.

Conditions

Applicable measurement view must be active.

Group

Sense commands

Syntax

`[SENSe]:PHVTime[:MEASview<y>]:PLOT {TRACE | OGRAM | SPLIT}`

`[SENSe]:PHVTime[:MEASview<y>]:PLOT?`

Arguments

`TRACE` displays time plot.

`OGRAM` displays ogram plot.

`SPLIT` displays both ogram and time plots.

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Returns

`{TRACE | OGRAM | SPLIT}`

Examples

`SENSe:PHVTime:MEASview1:PLOT TRACE` sets plot type to TRACE (Time plot).

[SENSe]:PHVTime[:MEASview<y>]:SELEcted:SOURce

Sets or queries the Source with respect to the Phaseversus Time Measurement View nuumber

Conditions

Measurement views: Phase versus Time with specified Measurement view number

Source cannot be changed when acquisition or analysis is in progress. Stop the acquisition and change the Source.

Group

Sense commands

Syntax

```
SENSe:PHVTime[:MEASview<y>]:SELEcted:SOURce {"Source 1" | "Source 2" | "Source 3" | "Source 4"}
SENSe: PHVTime[:MEASview<y>]:SELEcted:SOURce?
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Examples

SENSe:PHVTime:MEASview1:SELEcted:SOURce? queries or sets the Source for Phase versus Time Measurement View 1.

[SENSe]:PM:PHASe:OFFSet

Sets or queries the phase offset in the PM measurement.

Conditions

Measurement views: PM

Group

Sense commands

Syntax

```
[SENSe]:PM:PHASe:OFFSet <value>
[SENSe]:PM:PHASe:OFFSet?
```

Arguments

<value>::<NRf> specifies the phase offset. Range: –180 to +180 °.

Examples

SENSe:PM:PHASE:OFFSET 10deg sets the phase offset to 10 °.

[SENSe]:PM:PHASe:OFFSet:MARKer (No Query Form)

Sets the phase offset from the selected marker location in the PM measurement.

Conditions

Measurement views: PM

Group

Sense commands

Syntax

```
[SENSe]:PM:PHASe:OFFSet:MARKer
```

Arguments

None.

Examples

`SENSe:PM:PHASe:OFFSET:MARKER` sets the phase offset from the selected Marker position.

[SENSe]:PM:PHASe:SEARch:AUTO

Determines whether to detect the phase automatically or manually in the PM measurement.

Conditions

Measurement views: PM

Group

Sense commands

Syntax

```
[SENSe]:PM:PHASe:SEARch:AUTO { OFF | ON | 0 | 1 }
[SENSe]:PM:PHASe:SEARch:AUTO?
```

Arguments

`OFF` or `0` specifies that the phase offset is set manually using the command `[SENSe]:PM:PHASe:OFFSet`.

`ON` or `1` specifies that the phase is detected automatically.

Examples

`SENSe:PM:PHASE:SEARCH:AUTO ON` specifies that the phase is detected automatically.

[SENSe]:PNOise:AVERage:COUNt

Sets or queries the number of traces to combine for averaging in the phase noise measurement. This command is effective when `[SENSe]:PNOise:AVERage:ENABle` is set to ON.

Conditions

Measurement views: Phase noise

Group

Sense commands

Syntax

```
[SENSe]:PNOise:AVERage:COUNt <number>
[SENSe]:PNOise:AVERage:COUNt?
```

Arguments

`<number>::<NR1>` specifies the average count. Range: 2 to 10000.

Examples

`SENSe:PNOISE:AVERAGE:COUNT 64` sets the average count to 64.

[SENSe]:PNOise:AVERage:ENABle

Determines whether to enable averaging trace in the phase noise measurement.

Conditions

Measurement views: Phase noise

Group

Sense commands

Syntax

```
[SENSe]:PNOise:AVERage:ENABle {OFF|ON|0|1}
[SENSe]:PNOise:AVERage:ENABle?
```

Arguments

`OFF` disables averaging trace.

`ON` enables averaging trace.

Examples

`SENSe:PNOISE:AVERAGE:ENABLE ON` enables averaging trace.

[SENSe]:PNOise:CARRier:FREQuency:TRACk

Determines whether to enable or disable tracking the carrier frequency in the phase noise measurement.

Conditions

Measurement views: Phase noise

Group

Sense commands

Syntax

```
[SENSe]:PNOise:CARRier:FREQuency:TRACk {OFF|ON|0|1}
[SENSe]:PNOise:CARRier:FREQuency:TRACk?
```

Arguments

`OFF` or `0` disables tracking the carrier frequency.

`ON` or `1` enables tracking the carrier frequency.

Examples

`SENSe:PNOISE:CARRIER:FREQUENCY:TRACK ON` enables tracking the carrier frequency.

[SENSe]:PNOise:CARRier:THReshold

Sets or queries the threshold level to detect the carrier in the phase noise measurement.

Conditions

Measurement views: Phase noise

Group

Sense commands

Syntax

```
[SENSe]:PNOise:CARRier:THReshold <value>
[SENSe]:PNOise:CARRier:THReshold?
```

Arguments

`<value>::<NRf>` specifies the threshold level above which the input signal is determined to be a carrier. Range: -60 to 0 dBm.

Examples

`SENSe:PNOISE:CARRIER:THRESHOLD -25` sets the carrier threshold level to –25 dB.

[SENSe]:PNOise:CLEar:RESults (No Query Form)

Restarts the average process, clearing average data and counter.

Conditions

Measurement views: Phase noise

Group

Sense commands

Syntax

```
[SENSe]:PNOise:CLEar:RESults
```

Arguments

None

Examples

`SENSe:PNOISE:CLEAR:RESULTS` restarts the average process.

[SENSe]:PNOise:FREQuency:INTegration:OFFSet:STARt

Sets or queries the start offset frequency for integration in the phase noise measurement.

Conditions

Measurement views: Phase noise

Group

Sense commands

Syntax

```
[SENSe]:PNOise:FREQuency:INTegration:OFFSet:STARt <value>
[SENSe]:PNOise:FREQuency:INTegration:OFFSet:STARt?
```

Arguments

`<value>::<NRf>` specifies the start offset frequency for integration.

It must be less than the stop offset frequency. Range: 10 Hz to Stop Offset Frequency – 1 Hz.

Examples

`SENSe:PNOISE:FREQUENCY:INTEGRATION:OFFSET:START 100kHz` sets the start offset frequency for integration to 100 kHz.

[SENSe]:PNOise:FREQuency:INTegration:OFFSet:STOP

Sets or queries the stop offset frequency for integration in the phase noise measurement.

Conditions

Measurement views: Phase noise

Group

Sense commands

Syntax

```
[SENSe]:PNOise:FREQuency:INTegration:OFFSet:STOP <value>
[SENSe]:PNOise:FREQuency:INTegration:OFFSet:STOP?
```

Arguments

`<value>::<NRf>` specifies the stop offset frequency for integration.

It must be greater than the start offset frequency. Range: 11 Hz to 1 GHz.

Examples

`SENSe:PNOISE:FREQUENCY:INTEGRATION:OFFSET:STOP 100MHz` sets the stop offset frequency for integration to 100 MHz.

[SENSe]:PNOise:FREQuency:PLOT:OFFSet:STARt

Sets or queries the start offset frequency for plotting the phase noise trace.

Conditions

Measurement views: Phase noise

Group

Sense commands

Syntax

```
[SENSe]:PNOise:FREQuency:PLOT:OFFSet:STARt <value>
[SENSe]:PNOise:FREQuency:PLOT:OFFSet:STARt?
```

Arguments

`<value>::<NRf>` specifies the start offset frequency for the trace plot.

It must be less than the stop offset frequency. Range: 10 Hz to 100 MHz in a tenfold sequence.

Examples

`SENSe:PNOISE:FREQUENCY:PLOT:OFFSET:START 100kHz` sets the start offset frequency for plot to 100 kHz.

[SENSe]:PNOise:FREQuency:PLOT:OFFSet:STOP

Sets or queries the stop offset frequency for plotting the phase noise trace.

Conditions

Measurement views: Phase noise

Group

Sense commands

Syntax

```
[SENSe]:PNOise:FREQuency:PLOT:OFFSet:STOP <value>
[SENSe]:PNOise:FREQuency:PLOT:OFFSet:STOP?
```

Arguments

`<value>::<NRf>` specifies the stop offset frequency for plot.

It must be greater than the start offset frequency. Range: 100 Hz to 1 GHz in a tenfold sequence.

Examples

`SENSe:PNOISE:FREQUENCY:PLOT:OFFSET:STOP 100MHz` sets the stop offset frequency for plot to 100 MHz.

[SENSe]:PNOise:OPTimization

Sets or queries the method of optimizing the gain and input bandwidth in the phase noise measurement.

Conditions

Measurement views: Phase noise

Group

Sense commands

Syntax

[SENSe]:PNOise:OPTimization { DRANge | SPEed }

[SENSe]:PNOise:OPTimization?

Arguments

`DRANge` optimizes the gain and input bandwidth to maximize the dynamic range.

`SPEed` optimizes the gain and input bandwidth to speed the measurement.

Examples

`SENSe:PNOISE:OPTIMIZATION DRANge` optimizes the gain and input bandwidth to maximize the dynamic range.

[SENSe]:POWer:UNITs

Sets or queries the fundamental unit of power.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:POWer:UNITs {DBM|DBV|VOLTs|WATTs|DBUW|DBW|DBUV|DBMV|DBUA|DBUV_M|DBUA_M| AMPS}
```

```
[SENSe]:POWer:UNITs?
```

Arguments

The following table lists the arguments.

Power units

Argument	Power unit
DBM	dBm
DBV	dBV
VOLTs	Volts
WATTs	Watts
DBUW	dBµW
DBW	dBW
DBUV	dBµV
DBMV	dBmV
DBUA	dBµA
DBUV_M	dBµV/m
DBUA_M	dBµA/m
AMPS	Amps

NOTE: Select dBµV/m or dBµA/m unit when using an antenna table.

Examples

`SENSe:POWer:UNITs DBM` specifies the fundamental unit of power as dBm.

[SENSe]:PULSe[:MEASview<y>]:ANALyze:FDOMain:MEASurement:TIME:ALENgth

Sets or queries the absolute amount of time included in the Frequency-Domain Linearity type measurements.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:ANALyze:FDOMain:MEASurement:TIME:ALENgth <value>
```

[SENSe]:PULSe[:MEASview<y>]:ANALyze:FDOMain:MEASurement:TIME:ALENgth?

Arguments

<value>::<NRf> specifies the absolute time in seconds for the measurement time.

Range: 1 ns to 100 seconds

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf>

Examples

[SENSE]:PULSE:MEASview2:ANALYZE:FDOMAIN:MEASUREMENT:TIME:ALENGTH 1E-6 sets the measurement time to 1 µs.

[SENSe]:PULSe[:MEASview<y>]:ANALyze:FDOMain:MEASurement:TIME:METHod

Sets or queries the measurement time method for the Frequency-Domain Linearity type measurements.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Sense commands

Syntax

[SENSe]:PULSe[:MEASview<y>]:ANALyze:FDOMain:MEASurement:TIME:METHod { ABSolute | RELative }
[SENSe]:PULSe[:MEASview<y>]:ANALyze:FDOMain:MEASurement:TIME:METHod?

Arguments

ABSolute specifies the absolute time in seconds based on rising edge of the pulse.

RELative specifies the relative time in percent based on the pulse width and the center of the pulse.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf>

Examples

SENSe:PULSe:MEASview2:ANALyze:FDOMain:MEASurement:TIME:METHod REL sets the time method to Relative.

[SENSe]:PULSe[:MEASview<y>]:ANALyze:FDOMain:MEASurement:TIME:RLENgth

Sets or queries the relative amount of time included in the Frequency-Domain Linearity type measurements.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:ANALyze:FDOMain:MEASurement:TIME:RLENgth <value>
[SENSe]:PULSe[:MEASview<y>]:ANALyze:FDOMain:MEASurement:TIME:RLENgth?
```

Arguments

`<value>::<NRf>` specifies the relative time in percent for the measurement time based on the pulse width and centered of the pulse.

Range: 10% to 100%

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<NRf>`

Examples

`SENSe:PULSe:MEASview2:ANALyze:FDOMain:MEASurement:TIME:RLENgth 70` sets the measurement time to 70% of the pulse width.

[SENSe]:PULSe[:MEASview<y>]:ANALyze:FDOMain:MEASurement:TIME:RLEVel

Sets or queries the measurement reference level for Frequency-Domain Linearity measurements.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:ANALyze:FDOMain:MEASurement:TIME:RLEVel <value>
[SENSe]:PULSe[:MEASview<y>]:ANALyze:FDOMain:MEASurement:TIME:RLEVel?
```

Arguments

`<value>::<NRf>` specifies the reference level in percentage as the location on the voltage straight-line approximation of the pulse rising edge.

Range: 50% to 100%

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<NRf>`

Examples

`SENSe:PULSe:MEASview2:ANALyze:FDOMain:MEASurement:TIME:RLEVel 60` sets the reference level to 60% of the rising edge.

[SENSe]:PULSe[:MEASview<y>]:ANALyze:FDOMain:MEASurement:TIME:STARt

Sets or queries the start time for the Frequency-Domain Linearity type measurements. The start time is the amount of time wait after the rising-edge reference before starting to measure.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:ANALyze:FDOMain:MEASurement:TIME:STARt <value>
[SENSe]:PULSe[:MEASview<y>]:ANALyze:FDOMain:MEASurement:TIME:STARt?
```

Arguments

`<value>::<NRf>` specifies the start time in seconds from the reference level of the rising edge of the pulse.

Range: –100 ms to 100 ms

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<NRf>`

Examples

`SENSe:PULSe:MEASview2:ANALyze:FDOMain:MEASurement:TIME:STARt 1E-6` sets the start time to 1 µs.

[SENSe]:PULSe[:MEASview<y>]:ANALyze:IRESponse:CORRection:AMPLitude[:STATe]

Enables or disables the corrections for window and for delay of the highest side lobe in the Pulse Impulse Response measurements.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:ANALyze:IRESponse:CORRection:AMPLitude[:STATe] { OFF | ON | 0 | 1 }
[SENSe]:PULSe[:MEASview<y>]:ANALyze:IRESponse:CORRection:AMPLitude[:STATe]?
```

Arguments

OFF or 0 disables amplitude correction in Impulse Response measurements.

ON or 1 enables amplitude correction in Impulse Response measurements.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf>

Examples

SENSe:PULSe:MEASview2:ANALyze:IRESponse:CORRection:AMPLitude:STATe 1 enables amplitude correction.

[SENSe]:PULSe[:MEASview<y>]:ANALyze:IRESponse:KOTime

Sets or queries the Keep-out time in the Pulse Impulse Response measurements. This specifies the width of time around the main lobe that lobes within this time zone are not eligible to be the “highest side lobe.”

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:ANALyze:IRESponse:KOTime <value>
[SENSe]:PULSe[:MEASview<y>]:ANALyze:IRESponse:KOTime?
```

Arguments

<value>::<NRf> specifies the Keep-out time in seconds.

Range: 0 to 100 seconds.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf>

Examples

SENSe:PULSe:MEASview2:ANALyze:IRESponse:KOTime 0.001 sets the keep-out time to 1 ms.

[SENSe]:PULSe[:MEASview<y>]:ANALyze:IRESponse:MEASurement:TIME:ALENgth

Sets or queries the absolute amount of time included in the Impulse Response type measurements.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:ANALyze:IRESponse:MEASurement:TIME:ALENgth <value>
[SENSe]:PULSe[:MEASview<y>]:ANALyze:IRESponse:MEASurement:TIME:ALENgth?
```

Arguments

`<value>::<NRf>` specifies the absolute time in seconds for the measurement time.

Range: 1 ns to 100 seconds.

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<NRf>`

Examples

`SENSE:PULSE:MEASview2:ANALYZE:IRESPONSE:MEASUREMENT:TIME:ALEN 1E-6` sets the measurement time to 1 µs.

[SENSe]:PULSe[:MEASview<y>]:ANALyze:IRESponse:MEASurement:TIME: METHod

Sets or queries the measurement time method for the Impulse Response type measurements.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:ANALyze:IRESponse:MEASurement:TIME:METHod { ABSolute | RELative }
[SENSe]:PULSe[:MEASview<y>]:ANALyze:IRESponse:MEASurement:TIME:METHod?
```

Arguments

`ABSolute` specifies the absolute time in seconds based on rising edge of the pulse.

`RELative` specifies the relative time in percent based on the pulse width and the center of the pulse.

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<NRf>`

Examples

`SENSE:PULSE:MEASview2:ANALYZE:IRESPONSE:MEASUREMENT:TIME:METHOD REL` sets the time method to Relative.

[SENSe]:PULSe[:MEASview<y>]:ANALyze:IRESponse:MEASurement:TIME:RLENgth

Sets or queries the relative amount of time included in the Impulse Response type measurements.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:ANALyze:IRESponse:MEASurement:TIME:RLENgth <value>
[SENSe]:PULSe[:MEASview<y>]:ANALyze:IRESponse:MEASurement:TIME:RLENgth?
```

Arguments

`<value>::<NRf>` specifies the relative time in percent for the measurement time based on the pulse width and centered of the pulse.

Range: 10% to 100%.

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<NRf>`

Examples

`SENSE:PULSE:MEASview2:ANALYZE:IRESPONSE:MEASUREMENT:TIME:RLENGTH 70` sets the measurement time to 70% of the pulse width.

[SENSe]:PULSe[:MEASview<y>]:ANALyze:IRESponse:MEASurement:TIME: RLEVel

Sets or queries the measurement reference level for Impulse Response measurements.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:ANALyze:IRESponse:MEASurement:TIME:RLEVel <value>
[SENSe]:PULSe[:MEASview<y>]:ANALyze:IRESponse:MEASurement:TIME:RLEVel?
```

Arguments

`<value>::<NRf>` specifies the reference level in percentage as the location on the voltage straight-line approximation of the pulse rising edge.

Range: 50% to 100%.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf>

Examples

SENSE:PULSE:MEASview2:ANALYZE:IRESPONSE:MEASUREMENT:TIME:RLEVEL 60 sets the reference level to 60% of the rising edge.

[SENSe]:PULSe[:MEASview<y>]:ANALyze:IRESponse:MEASurement:TIME:STARt

Sets or queries the start time for the Impulse Response type measurements. The start time is the amount of time wait after the rising-edge reference before starting to measure.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:ANALyze:IRESponse:MEASurement:TIME:STARt <value>
[SENSe]:PULSe[:MEASview<y>]:ANALyze:IRESponse:MEASurement:TIME:STARt?
```

Arguments

<value>::<NRf> specifies the start time in seconds from the reference level of the rising edge of the pulse.

Range: −100 ms to 100 ms.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf>

Examples

SENSE:PULSE:MEASview2:ANALYZE:IRESPONSE:MEASUREMENT:TIME:START 1E-6 sets the start time to 1 µs.

[SENSe]:PULSe[:MEASview<y>]:ANALyze: LEVel

Sets or queries how to determine the 50% level for the pulsed RF measurements.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:ANALyze:LEVel { VOLTage | POWer }
[SENSe]:PULSe[:MEASview<y>]:ANALyze:LEVel?
```

Arguments

`VOLTage` uses −6 dB to determine the 50% level.

`POWer` uses −3 dB to determine the 50% level.

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<NRf>`

Examples

`SENSe:PULSe:MEASview2:ANALyze:LEVel POWer` sets RF measurements to use −3 dB to determine the 50% level.

[SENSe]:PULSe[:MEASview<y>]:ANALyze:LEVel:FIFTy

Sets or queries how to determine the 50% level for the pulsed RF measurements. This command is equivalent to the `[SENSe]:PULSe[:MEASview<y>]:ANALyze:LEVel` command.

Conditions

Measurement views: Pulse statistics, Pulse table, Pulse trace

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:ANALyze:LEVel:FIFTy { VOLTage | POWer }
[SENSe]:PULSe[:MEASview<y>]:ANALyze:LEVel:FIFTy?
```

Arguments

`VOLTage` uses −6 dB to determine the 50% level.

`POWer` uses −3 dB to determine the 50% level.

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Examples

`SENSe:PULSE:MEASview2:ANALYZE:LEVEL:FIFTY POWer` sets RF measurements to use -3 dB to determine the 50% level.

[SENSe]:PULSe[:MEASview<y>]:ANALyze:LEVel:HUNDred

Sets or queries how to determine the 100% level in the pulsed RF measurements. This command is equivalent to the `[SENSe]:PULSe:ANALyze:LEVel` command.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:ANALyze:LEVel:HUNDred { AVERage | INDependent }
[SENSe]:PULSe[:MEASview<y>]:ANALyze:LEVel:HUNDred?
```

Arguments

`AVERage` uses the average amplitude calculated for the pulse-on as the 100% reference to measure the rise and fall times.

`INDependent` uses the amplitudes at the beginning and end of the pulse-on as the 100% references to measure the rise and fall times, respectively

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<NRf>`

Examples

`SENSe:PULSe:MEASview2:ANALyze:LEVel:HUNDRED AVERage` uses pulse average amplitude to set the 100% level.

[SENSe]:PULSe[:MEASview<y>]:ANALyze:MEASurement:TIME:TYPE?

Sets or Queries the Measurement(s) parameter from the Control panel settings of Define tab which is common across the displays Pulse Statistics, Pulse Table, Pulse Trace, Pulse Cumulative Histogram, Pulse Cumulative Statistics.

Conditions

Measurement views: Pulse Statistics, Pulse Table, Pulse Trace, Pulse Cumulative Histogram, Pulse Cumulative Statistics

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:ANALyze:MEASurement:TIME:TYPE ?
```

Arguments

`<arg1> :: {FDL | IMPR | FPD | RIPP | OVER}` sets the control panel Measurement(s) parameter of define tab to Freq-domain Linearity or Impulse Response or F P2R Diff, F P2P Diff, Freq Abs, Phase P2R Diff, Phase P2P Diff or Ripple or Overshoot

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<string>`

Examples

`SENSe:PULSE:ANALYZE:MEASview1:MEASUREMENT:TIME:TYPE FDL` sets the control panel Measurement(s) parameter under Define tab to Freq-domain Linearity.

[SENSe]:PULSe[:MEASview<y>]:ANALyze:MEASurement:TIME:TYPE

Sets or queries the Measurement(s) parameter for the Define tab of the Settings control panel, which is common across the Pulse Statistics, Pulse Table, Pulse Trace, Pulse Cumulative Histogram, and Pulse Cumulative Statistics displays.

Conditions

Measurement views: All Pulsed RF displays (except Pulse-Ogram)

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:ANALyze:MEASurement:TIME:TYPE { FDL | IMPR | FPD | RIPP | OVER }
[SENSe]:PULSe[:MEASview<y>]:ANALyze:MEASurement:TIME:TYPE ?
```

Arguments

`FDL` is Freq-domain Linearity

`IMPR` is Impulse Response

`FPD` is F P2R Diff

`RIPP` is ripple

`OVER` is overshoot

Freq Abs, Phase P2R Diff, Phase P2P Diff

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<string>`

Examples

`SENSe:PULSE:MEASview1:ANALYZE:MEASUREMENT:TIME:TYPE FDL` sets the Measurement(s) parameter in the Define tab of the Settings control panel to Freq-domain Linearity.

[SENSe]:PULSe[:MEASview<y>]:ANALyze:OVERshoot:MEASurement:TIME:ALENgth

Sets or queries the absolute amount of time included in the Overshoot measurements.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:ANALyze:OVERshoot:MEASurement:TIME:ALENgth <value>
[SENSe]:PULSe[:MEASview<y>]:ANALyze:OVERshoot:MEASurement:TIME:ALENgth?
```

Arguments

<value>::<NRf> specifies the absolute time in seconds for the measurement time.

Range: 1 ns to 100 seconds

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf>

Examples

[SENSE]:PULSE:MEASview2:ANALYZE:OVERSHOOT:MEASUREMENT:TIME:ALENGTH 1E-6 sets the measurement time to 1 µs.

[SENSe]:PULSe[:MEASview<y>]:ANALyze:OVERshoot:MEASurement:TIME: METHod

Sets or queries the measurement time method for the Overshoot measurements.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:ANALyze:OVERshoot:MEASurement:TIME:METHod { ABSolute | RELative }
[SENSe]:PULSe[:MEASview<y>]:ANALyze:OVERshoot:MEASurement:TIME:METHod?
```

Arguments

ABSolute specifies the absolute time in seconds based on rising edge of the pulse.

RELative specifies the relative time in percent based on the pulse width and the center of the pulse.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf>

Examples

SENSE:PULSE:MEASview2:ANALYZE:OVERSHOOT:MEASUREMENT:TIME:METHOD REL sets Overshoot measurements to use the relative time method.

[SENSe]:PULSe[:MEASview<y>]:ANALyze:OVERshoot:MEASurement:TIME:RLENgth

Sets or queries the relative amount of time included in the Overshoot measurements.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:ANALyze:OVERshoot:MEASurement:TIME:RLENgth <value>
[SENSe]:PULSe[:MEASview<y>]:ANALyze:OVERshoot:MEASurement:TIME:RLENgth?
```

Arguments

<value>::<NRf> specifies the relative time in percent for the measurement time based on the pulse width and centered of the pulse.

Range: 0.01% to 100%.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf>

Examples

SENSE:PULSE:MEASview2:ANALYZE:OVERSHOOT:MEASUREMENT:TIME:RLENGTH 70 sets the measurement time to 70% of the pulse width.

[SENSe]:PULSe[:MEASview<y>]:ANALyze:OVERshoot:MEASurement:TIME: RLEVel

Sets or queries the measurement reference level for the Overshoot measurements.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:ANALyze:OVERshoot:MEASurement:TIME:RLEVel <value>
[SENSe]:PULSe[:MEASview<y>]:ANALyze:OVERshoot:MEASurement:TIME:RLEVel?
```

Arguments

<value>::<NRf> specifies the reference level in percentage as the location on the voltage straight-line approximation of the pulse rising edge.

Range: 50% to 100%.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf>

Examples

SENSE:PULSE:MEASview2:ANALYZE:OVERSHOOT:MEASUREMENT:TIME:RLEVEL 60 sets the reference level to 60% of the rising edge.

[SENSe]:PULSe[:MEASview<y>]:ANALyze:OVERshoot:MEASurement:TIME:STARt

Sets or queries the start time for the Overshoot measurements. The start time is the amount of time wait after the rising-edge reference before starting to measure.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:ANALyze:OVERshoot:MEASurement:TIME:STARt <value>
[SENSe]:PULSe[:MEASview<y>]:ANALyze:OVERshoot:MEASurement:TIME:STARt?
```

Arguments

<value>::<NRf> specifies the start time in seconds from the reference level of the rising edge of the pulse.

Range: –100 ms to 100 ms.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf>

Examples

SENSE:PULSE:MEASview2:ANALYZE:OVERSHOOT:MEASUREMENT:TIME:START 1E-6 sets the start time to 1 µs to 60% of the rising edge.

[SENSe]:PULSe[:MEASview<y>]:ANALyze:PMLocation

Sets or queries the phase measurement location (the position along the pulse tops where the phase is measured) in the pulse-pulse phase measurement.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:ANALyze:PMLocation <value>
[SENSe]:PULSe[:MEASview<y>]:ANALyze:PMLocation?
```

Arguments

<value>::<NRf> specifies the pulse-pulse phase measurement location.

Range: 5 ns to 100 ms.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf>

Examples

SENSe:PULSe:MEASview2:ANALyze:PMLocation 1.5ms sets the phase measurement location to 1.5 ms.

[SENSe]:PULSe[:MEASview<y>]:ANALyze:POINt:LOCation

Sets or queries the point location method in the pulsed RF measurements.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:ANALyze:POINt:LOCation { AVERage | INDependent }
[SENSe]:PULSe[:MEASview<y>]:ANALyze:POINt:LOCation?
```

Arguments

AVERage uses the average amplitude calculated for the pulse-on as the 100% reference to measure the rise and fall times.

INDependent uses the amplitudes at the beginning and end of the pulse-on as the 100% references to measure the rise and fall times, respectively.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf>

Examples

[SENSe]:PULSe:MEASview2:ANALyze:POINt:LOCation AVERage uses pulse average amplitude to locate points.

[SENSe]:PULSe[:MEASview<y>]:ANALyze:RFALI

Sets or queries the threshold levels to measure the rise/fall time.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:ANALyze:RFALl { WIDE | NARRow }
[SENSe]:PULSe[:MEASview<y>]:ANALyze:RFALl?
```

Arguments

`WIDE` selects 10 - 90% to measure the rise/fall time.

`NARRow` selects 20 - 80% to measure the rise/fall time.

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<NRf>`

Examples

`SENSe:PULSe:MEASview2:ANALyze:RFALL WIDE` selects 10 - 90% to measure the rise/fall time.

[SENSe]:PULSe[:MEASview<y>]:ANALyze:RIPPle

Sets or queries the ripple portion of the pulse top (how much of the beginning and end of the pulse top to exclude from the ripple calculation).

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:ANALyze:RIPPle <value>
[SENSe]:PULSe[:MEASview<y>]:ANALyze:RIPPle?
```

Arguments

`<value>::<NRf>` specifies the ripple portion of the pulse top.

Range: 10 to 100% in 1% steps.

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<NRf>`

Examples

`SENSe:PULSe:MEASview2:ANALyze:RIPPle 30` sets the ripple portion to 30% of the pulse top.

[SENSe]:PULSe:CARRier:FREQuency? (Query Only)

Queries the carrier frequency.

Conditions

Measurement views: Pulse table

Group

Sense commands

Syntax

```
[SENSe]:PULSe:CARRier:FREQuency?
```

Related Commands

[SENSe]:PULSe:CARRier:OFFSet (on page 2286)

Arguments

None

Returns

<NRf> indicates the actual carrier frequency in Hz.

Examples

SENSe:PULSE:CARRIER:FREQUENCY? might return 2.7000001690E+9, indicating that the pulse carrier frequency is 2.700000169 GHz.

[SENSe]:PULSe[:MEASview<y>]:CARRier:OFFSet

Sets or queries the carrier frequency offset.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:CARRier:OFFSet <value>
[SENSe]:PULSe[:MEASview<y>]:CARRier:OFFSet?
```

Related Commands

[SENSe]:PULSe[:MEASview<y>:CARRier:FREQuency? (on page 2285)

Arguments

<value>::<NRf> specifies the carrier frequency offset.

Range: -50 kHz to +50 kHz.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf>

Examples

SENSe:PULSe:MEASview2:CARRier:OFFSet 2.5kHz sets the offset frequency to 2.5 kHz to the carrier.

[SENSe]:PULSe[:MEASview<y>]:CARRier:SEARch

Sets or queries how to detect the carrier in the pulsed RF measurements.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:CARRier:SEARch { AUTO | MANual}
[SENSe]:PULSe[:MEASview<y>]:CARRier:SEARch?
```

Arguments

AUTO specifies that the carrier is detected automatically.

MANual specifies that the carrier frequency offset is set manually, using the [SENSe]:PULSe:CARRier:OFFSet command.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf>

Examples

SENSe:PULSe:MEASview2:CARRier:SEARch AUTO specifies that the carrier is detected automatically.

[SENSe]:PULSe:CUMulative:HISTogram[:MEASview<y>]:BIN:COUNt

Sets the Bins value in Histogram settings table in the Cumulative Histogram display.

Conditions

Measurement views: Cumulative Histogram with the specified Measurement View Number

Group

Sense commands

Syntax

```
[SENSe]:PULSe:CUMulative:HISTogram[:MEASview<y>]:BIN:COUNt <arg1>
```

Arguments

<arg1> <arg1> 100 sets the bin's value in the histogram settings table to 100 which is under Histogram Settings table in Pulse Cumulative Histogram control panel.

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf1>

Examples

SENSe:PULSE:CUMULATIVE:HISTOGRAM:MEASview1:BIN:COUNT 100 sets the bin's value in the histogram table to 100.

[SENSe]:PULSe:CUMulative:HISTogram[:MEASview<y>]:CLEar:RESults (Query Only)

Clears all scalar results in the Cumulative Histogram display.

Conditions

Measurement views: Cumulative Histogram with the specified Measurement View Number

Group

Sense commands

Syntax

[SENSe]:PULSe:CUMulative:HISTogram[:MEASview<y>]:CLEar:RESults

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Examples

SENSe:PULSE:CUMULATIVE:HISTOGRAM:MEASview1:CLEAR:RESULTS clears the result in Cumulative Histogram display.

[SENSe]:PULSe:CUMulative:HISTogram[:MEASview<y>]:MODE

Sets the Histogram settings mode to Auto or Manual in the Cumulative Histogram display.

Conditions

Measurement views: Cumulative Histogram with the specified Measurement View Number

Group

Sense commands

Syntax

SENSe:PULSe:CUMulative:HISTogram[:MEASview<y>]:MODE <arg1>

Arguments

<arg1> <arg1> AUTO or MAN sets the histogram settings mode to Auto or Manual which is under Histogram Settings table in Pulse Cumulative Histogram control panel.

Returns

<string>

Examples

SENSe:PULSE:CUMULATIVE:HISTOGRAM:MEASview1:MODE AUTO sets the histogram settings mode to Auto.

[SENSe]:PULSe:CUMulative:HISTogram[:MEASview<y>]:ORDinate (Query Only)

Sets or queries the Histogram readout in the Cumulative Histogram display.

Conditions

Measurement views: Cumulative Histogram with the specified Measurement View Number

Group

Sense commands

Syntax

```
SENSe:PULSe:CUMulative:HISTogram[:MEASview<y>]:ORDinate <arg1>
SENSe:PULSe:CUMulative:HISTogram[:MEASview<y>]:ORDinate?
```

Arguments

`<arg1>` `<arg1> COUN` or `PERC` sets the histogram readout to Count or % which is under Prefs tab in Pulse Cumulative Histogram control panel.

The parameter `<y>` = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<string>`

Examples

`SENSe:PULSE:CUMULATIVE:HISTOGRAM:MEASview1:ORDINATE ORDINATE COUN` sets the histogram readout to Count.

[SENSe]:PULSe:CUMulative:HISTogram[:MEASview<y>]:OUTLier:ORDinate

Sets the outlier readout in the Cumulative Histogram display.

Conditions

Measurement views: Cumulative Histogram with the specified Measurement View Number

Group

Sense commands

Syntax

```
SENSe:PULSe:CUMulative:HISTogram[:MEASview<y>]:OUTLIER:ORDINATE <arg1>
```

Arguments

`<arg1>` `<arg1> COUN` or `PERC` sets the outlier readout to Count or % which is under Prefs tab in Pulse cumulative histogram control panel.

Returns

`<string>`

Examples

SENSe:PULSE:CUMULATIVE:HISTOGRAM:OUTLIER:MEASview1:ORDINATE COUN sets the outlier readout to Count.

[SENSe]:PULSe:CUMulative:HISTogram[:MEASview<y>]:SETtings:APPly (No Query Form)

Applies the new settings to the Cumulative Histogram display.

Conditions

Measurement views: Cumulative Histogram with the specified Measurement View Number

Group

Sense commands

Syntax

[SENSe]:PULSe:CUMulative:HISTogram[:MEASview<y>]:SETtings:APPly

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Examples

SENSe:PULSE:CUMULATIVE:HISTOGRAM:MEASview1:SETTINGS:APPLY applies new settings to the Cumulative Histogram display under Histogram Settings table in Pulse Cumulative Histogram control panel.

[SENSe]:PULSe:CUMulative:HISTogram[:MEASview<y>]:XLIMit

Sets the Xlimit in Histogram settings table in the Cumulative Histogram display.

Conditions

Measurement views: Cumulative Histogram with the specified Measurement View Number

Group

Sense commands

Syntax

[SENSe]:PULSe:CUMulative:HISTogram[:MEASview<y>]:XLIMit <arg1>,<arg2>,<arg3>,<arg4>

Arguments arg1

arg1 can have one of the following result types:

{ AVERage| PPOWer | ATX | WIDTh | RISE | FALL | RINTerval | RRATe | DUTPct | DUTRatio | RIPPle | RIPDb | DROop | DRODb | OVERshoot | OVEDb | PPPHase | PPFRequency | RMSFreqerror | MFReqerror | RMSPherror | MPHerror | FRDeviation | FDELta | PHDeviation | IRAMplitude | TIME | IRTime | PPPD | PPFD | FABS}

arg2

0 unchecks the box

1 checks the box.

Where

<arg1> is the Pulse result type.

<arg2> is the active flag which enables the limit state ON.

<arg3> is the minimum value.

<arg4> is the maximum value.

Examples

SENSe:PULSE:CUMULATIVE:HISTOGRAM:MEASview1:XLIMIT AVG,0,10,65 sets the Pulse result type, Active flag, Minimum value and Maximum value in the histogram table.

SENSe:PULSe:CUMulative:HISTogram[:MEASview<y>]:SELEcted:SOURce

Returns the Source number of specified Cumulative Histogram trace

Conditions

Measurement views: Cumulative Histogram with the specified Measurement View Number

Source cannot be changed when acquisition or analysis is in progress. Stop the acquisition and change the Source.

Group

Sense commands

Syntax

```
SENSe:PULSe:CUMulative:HISTogram[:MEASview<y>]:SELEcted:SOURce {"Source 1" |"Source 2" |"Source 3"
    |"Source 4"}
SENSe:PULSe:CUMulative:HISTogram[:MEASview<y>]:SELEcted:SOURce? (QUERY)
```

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf> Source Number of the specified Spectrogram tace

Examples

SENSe:PULSe:CUMulative:HISTogram:MEASview1:SELEcted:SOURce? might return Source 2 for the trace of Cumulative Histogram measurement view 1.

[SENSe]:PULSe:CUMulative:STATistics[:MEASview<y>]:CLEar:RESults

Clears all scalar results in the Cumulative Statistics display.

Conditions

Measurement views: Pulse Cumulative Statistics

Group

Sense commands

Syntax

```
[SENSe]:PULSe:CUMulative:STATistics[:MEASview<y>]:CLEar:RESults
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Examples

`SENSe:PULSe:CUMulative:STATistics:MEASview1:CLEar:RESults` clears the result in Cumulative Statistics display.

[SENSe]:PULSe:CUMulative:STATistics[:MEASview<y>]:SELEcted:SOURce

Sets or queries the selected source for Pulse Cumulative statistics measurment.

Conditions

Measurement views: Pulse Cumulative Statistics

Source cannot be changed when acquisition or analysis is in progress. Stop the acquisition and change the Source.

Group

Sense commands

Syntax

```
[SENSe]:PULSe:CUMulative:STATistics[:MEASview<y>]:SELEcted:SOURce {"Source 1" |"Source 2" |"Source
   3" |"Source 4"}
[SENSe]:PULSe:CUMulative:STATistics[:MEASview<y>]:SELEcted:SOURce?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Returns

```
{"Source 1" | "Source 2" | "Source 3" | "Source 4"}
```

Examples

`SENSe:PULSe:CUMulative:STATistics:MEASview1:SELEcted:SOURce?` might return source `3`, which indicates the channel of measurement source for RF data type.

[SENSe]:PULSe[:MEASview<y>]:DETect:MEASurement

Determines whether or not to set the maximum number of pulses to measure within the analysis time.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:DETect:MEASurement { OFF | ON | 0 | 1 }
```

[SENSe]:PULSe[:MEASview<y>]:DETect:MEASurement?

Arguments

OFF or 0 measures all pulses (max. 1000) in the analysis time.

ON or 1 specifies that the maximum number of pulses is set manually, using the [SENSe]:PULSe:DETect:NUMBer command.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf>

Examples

SENSe:PULSe:MEASview2:DETect:MEASurement ON specifies that the maximum number of pulses is set manually.

[SENSe]:PULSe[:MEASview<y>]:DETect:NUMBer

Sets or queries the maximum number of pulses to measure within the analysis time when [SENSe]:PULSe:DETect:MEASurement is On.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Sense commands

Syntax

[SENSe]:PULSe[:MEASview<y>]:DETect:NUMBer <value>
[SENSe]:PULSe[:MEASview<y>]:DETect:NUMBer?

Arguments

<value>::<NRf> specifies the maximum number of pulses to measure within the analysis time. Range: 1 to 1000.

If the analysis time contains fewer pulses than this number, all of these are measured.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf>

Examples

SENSe:PULSe:MEASview2:DETect:NUMBer 850 sets the maximum number of pulses to 850.

[SENSe]:PULSe[:MEASview<y>]:DETect:NUMBer:THReshold

Sets or queries the power threshold to detect pulses.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:DETect:NUMBer:THReshold <value>
[SENSe]:PULSe[:MEASview<y>]:DETect:NUMBer:THReshold?
```

Arguments

`<value>::<NRf>` specifies the power threshold to detect pulses.

Range: -70 to 0 dB.

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<NRf>`

Examples

`SENSe:PULSe:MEASview2:DETect:NUMBer:THReshold -20` sets the power threshold to -20 dB.

[SENSe]:PULSe[:MEASview<y>]:DETect:TIMe[:THReshold]

Sets or queries the minimum off-time between pulses.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:DETect:TIMe[:THReshold] <value>
[SENSe]:PULSe[:MEASview<y>]:DETect:TIMe[:THReshold]?
```

Arguments

`<value>::<NRf>` specifies the minimum off-time between pulses.

Range: 1 ns to 100 ms.

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<NRf>`

Examples

`SENSe:PULSe:MEASview2:DETect:TIMe:THReshold 1.5ms` sets the time threshold to 1.5 ms.

[SENSe]:PULSe[:MEASview<y>]:FILTer:{BANDwidth|BWIDth}

[SENSe]:PULSe[:MEASview<y>]:FILTer:{BANDwidth | BWIDth}

Sets or queries the filter or acquisition bandwidth when `[SENSe]:PULSe:FILTer:MEASurement` is set to `GAUSsian` or `NONE` (No filter), respectively.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:FILTer:{BANDwidth|BWIDth} <value>
[SENSe]:PULSe[:MEASview<y>]:FILTer:{BANDwidth|BWIDth}?
```

Related Commands

[SENSe]:PULSe:FILTer:MEASurement (on page 2295)

Arguments

`<value>::<NRf>` specifies the filter/acquisition bandwidth depending the `[SENSe]:PULSe:FILTer:MEASurement` setting. The table below shows the setting range. You can enter any value, but it is rounded up to the next valid number.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

[SENSe]:PULSe:FILTer:MEASurement	Range
GAUSsian	100 Hz to 20 MHz (Standard) / 100 Hz to 55 MHz (Option 110) in 1-2-3-5 sequence.
NONE	152, 305, 610, 1.22 k, 2.44 k, 4.88 k, 9.76 k, 19.3 k, 39 k, 78 k, 156 k, 312 k, 625 k, 1.25 M, 2.5 M, 5 M, 10 M, 20 M, 40 MHz, and optionally 60 M and 110 MHz (Option 110).

Returns

`<NRf>`

Examples

`SENSe:PULSe:MEASview2:FILTer:BANDwidth 10MHz` sets the filter bandwidth to 10 MHz.

[SENSe]:PULSe[:MEASview<y>]:FILTer:MEASurement

Sets or queries the measurement filter in the pulsed RF measurements.

Conditions

Measurement views: Pulse statistics, Pulse table, Pulse trace

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:FILTer:MEASurement { GAUSsian | NONE | MAXRtbw}
[SENSe]:PULSe[:MEASview<y>]:FILTer:MEASurement?
```

Related Commands

[SENSe]:PULSe[:MEASview<y>]:FILTer {BANDwidth|BWIDth} (on page 2295)

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

GAUSsian uses the Gaussian filter in the pulsed RF measurements. Use the [SENSe]:PULSe[:MEASview<y>]:FILTer:{BANDwidth|BWIDth} command to set the filter bandwidth.

NONE uses no filter. Use the [SENSe]:PULSe[:MEASview<y>]:FILTer:{BANDwidth|BWIDth} command to set the acquisition bandwidth.

MAXRtbw uses no filter. The acquisition bandwidth is fixed to the maximum real-time bandwidth: 40 MHz (Standard) or 110 MHz (Option 110).

Examples

SENSe:PULSE:MEASview1:FILTER:MEASUREMENT GAUSsian uses the Gaussian filter in the pulsed RF measurements.

[SENSe]:PULSe[:MEASview<y>]:FREFerence:AUTO

Determines whether to estimate the pulse frequency reference automatically or manually in the pulsed RF measurements.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:FREFerence:AUTO { OFF | ON | 0 | 1 }
[SENSe]:PULSe[:MEASview<y>]:FREFerence:AUTO?
```

Arguments

OFF or 0 estimates the frequency reference manually. Use the [SENSe]:PULSe:FREFerence:OFFSet command to set the frequency offset.

ON or 1 estimates the frequency reference automatically.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf>

Examples

SENSe:PULSe[:MEASview2:FREFerence:AUTO ON specifies that the maximum number of pulses is set manually.

[SENSe]:PULSe[:MEASview<y>]:FREFerence:CSLope

Sets or queries the Chirp Slope.

Chirp Slope is used when the Modulation Type is set to Linear Chirp. When set, this also changes the Chirp Slope to manual mode. When chirp slope is in auto mode, the query returns the estimated chirp slope.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:FREFerence:CSLope <value>
[SENSe]:PULSe[:MEASview<y>]:FREFerence:CSLope?
```

Arguments

`<value>`: = NRf specifies the chirp slope in Hz/µs.

Range: -100 GHz/µs to 100 GHz/µs.

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<NRf>`

Examples

`SENSe:PULSe:MEASview2:FREFerence:CSLope 1E6` sets the chirp slope to 1 MHz/µs.

[SENSe]:PULSe[:MEASview<y>]:FREFerence:CSLope:AUTO

Sets or queries enabling automatic Chirp Slope estimating in the pulsed RF measurements.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:FREFerence:CSLope:AUTO { OFF | ON | 0 | 1 }
[SENSe]:PULSe[:MEASview<y>]:FREFerence:CSLope:AUTO?
```

Arguments

`OFF` or `0` estimates the chirp slope manually. Use the `[SENSe]:PULSe[:MEASview<y>]:FREFerence:CSLope` command to set the chirp slope.

`ON` or `1` estimates the chirp slope automatically.

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf>

Examples

SENSe:PULSe:MEASview2:FREFerence:CSLope:AUTO ON sets the chirp slope to auto mode.

[SENSe]:PULSe[:MEASview<y>]:FREFerence:OFFSet

Sets or queries the frequency reference offset. This command is valid when [SENSe]:PULSe[:MEASview<y>]:FREFerence:AUTO is set to OFF.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:FREFerence:OFFSet <value>
[SENSe]:PULSe[:MEASview<y>]:FREFerence:OFFSet?
```

Arguments

<value>::<NRf> specifies the frequency reference offset.

Range: -50 kHz to +50 kHz.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf>

Examples

SENSe:PULSe:MEASview2:FREFerence:OFFSet 2.5kHz sets the frequency offset to 2.5 kHz to the reference.

[SENSe]:PULSe[:MEASview<y>]:MODulation:TYPE

Sets or queries the modulation type in the pulsed RF measurements. This command is equivalent to the [SENSe]:REANalyze:NEXT command.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:MODulation:TYPE { CWConst | CWCHange | LCHirp | OTHer}
[SENSe]:PULSe[:MEASview<y>]:MODulation:TYPE?
```

Arguments

`CWConst` selects the CW (continuous wave) with constant phase.

`CWCHange` selects the CW (continuous wave) with changing phase.

`LCHirp` selects the linear chirp.

`OTHer` selects other.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf>

Examples

`SENSe:PULSe:MEASview2:MODulation:TYPE CWCHang` selects the CW with changing phase as the modulation type.

[SENSe]:PULSe[:MEASview<y>]:STATistics:SELEcted:SOURce

Sets or queries selected source for pulse statistics measurement.

Conditions

Measurement views: Pulse Statistics

Source cannot be changed when acquisition or analysis is in progress. Stop the acquisition and change the Source.

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:STATistics:SELEcted:SOURce {"Source 1" | "Source 2" | "Source 3" |
   "Source 4" }
[SENSe]:PULSe[:MEASview<y>]:STATistics:SELEcted:SOURce?
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf>

Examples

`SENSe:PULSe:MEASview2:STATistics:SELEcted:SOURce?` queries the source of the Pulse Statistics Measurement View 2.

[SENSe]:PULSe:OGRAm[:MEASview<y>]: ANALysis:FVTime:FILTer:SHAPe

Sets or queries the type of Filter shape from Analysis tab of Frequency vs Time from the Pulse-Ogram display.

Conditions

Measurement views: Pulse-Ogram

Group

Sense commands

Syntax

```
[SENSe]:PULSe:OGRAm[:MEASview<y>]:ANALysis:FVTime:FILTer:SHAPe <arg1>
[SENSe]:PULSe:OGRAm[:MEASview<y>]:ANALysis:FVTime:FILTer:SHAPe?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

<arg1> is KAIS or UNIF or BH4B or FLAT or HAMM

Where

UNIF is Uniform (none).

BH4B is Blackman-Harris 4B.

FLAT is Flat - Top (CW ampl).

KAIS is Kaiser (RBW).

HAMM is Hamming.

Returns

<string> KAIS or UNIF or BH4B or FLAT or HAMM

Examples

SENSe:PULSe:OGRAm:MEASview1:ANALysis:FVTime:FILTer:SHAPe KAIS sets the Analysis Filter shape to Kaiser(RBW).

[SENSe]:PULSe:OGRAm[:MEASview<y>]:ANALysis:FVTime:SPECtrum:ANALysis:REGion

Sets or queries the Spectrum Analysis region from Analysis tab of Frequency vs Time in the Pulse-Ogram display.

Conditions

Measurement views: Pulse-Ogram

Group

Sense commands

Syntax

```
[SENSe]:PULSe:OGRAm[:MEASview<y>]:ANALysis:FVTime:SPECtrum:ANALysis:REGion <arg1>
[SENSe]:PULSe:OGRAm[:MEASview<y>]:ANALysis:FVTime:SPECtrum:ANALysis:REGion?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

<arg1> is ATIM or POT or IND

Where

ATIM is Use Analysis Time settings.

POT is Pulse ON Time.

IND is Independent.

Returns

<string> ATIM or POT or IND

Examples

SENSe:PULSe:OGRAm:MEASview1:ANALysis:FVTime:SPECtrum:ANALysis:REGion ATIM sets the Spectrum Analysis region to Use Analysis Time settings option.

[SENSe]:PULSe:OGRAm[:MEASview<y>]:SELEcted:SOURce

Returns the Source number of specified pulso-gram trace.

Conditions

Measurement views: Pulse-Ogram

Source cannot be changed when acquisition or analysis is in progress. Stop the acquisition and change the Source.

Group

Sense commands

Syntax

```
[SENSe]:PULSe:OGRAm[:MEASview<y>]:SELEcted:SOURce {"Source 1" | "Source 2" | "Source 3" | "Source
    4" }
[SENSe]:PULSe:OGRAm[:MEASview<y>]:SELEcted:SOURce?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Returns

{"Source 1" | "Source 2" | "Source 3" | "Source 4" }

Examples

SENSe:PULSe:OGRAm:MEASview1:SELEcted:SOURce? queries the source of the PulseOgram measurement view 1.

[SENSe]:PULSe:OGRAm[:MEASview<y>]:ANALysis:FVTime:ONTime:LENgth

Sets or queries the OnTime value from Analysis tab of Frequency vs Time in the Pulse-Ogram display.

Conditions

Measurement views: Pulse-Ogram

Group

Sense commands

Syntax

```
[SENSe]:PULSe:OGRAm[:MEASview<y>]:ANALysis:FVTime:ONTime:LENgth <value>
[SENSe]:PULSe:OGRAm[:MEASview<y>]:ANALysis:FVTime:ONTime:LENgth?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

<value> = <NRf>

Returns

<NRf> specifies the ON time in seconds.

Examples

SENSe:PULSe:OGRAm:MEASview<y>:ANALysis:FVTime:ONTime:LENgth? might return 10E-6s.

[SENSe]:PULSe:OGRAm[:MEASview<y>]:ANALysis:FVTime:INDependent:STARt

Sets or queries the independent spectrum start time from Analysis tab of Frequency vs Time in the Pulse-Ogram display.

Conditions

Measurement views: Pulse-Ogram

Group

Sense commands

Syntax

```
[SENSe]:PULSe:OGRAm[:MEASview<y>]:ANALysis:FVTime:INDependent:STARt <value>
[SENSe]:PULSe:OGRAm[:MEASview<y>]:ANALysis:FVTime:INDependent:STARt?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1.

<value> = <NRf>

Returns

<NRf> specifies the independent spectrum start time in seconds.

Examples

SENSe:PULSe:OGRAm:MEASview1:ANALysis:FVTime:INDependent:STARt? might return 10E-6s.

[SENSe]:PULSe:OGRAm[:MEASview<y>]:ANALysis:FVTime:INDependent:LENgth

Sets or queries the independent spectrum length from Analysis tab of Frequency vs Time in the Pulse-Ogram display.

Conditions

Measurement views: Pulse-Ogram

Group

Sense commands

Syntax

```
[SENSe]:PULSe:OGRAm[:MEASview<y>]:ANALysis:FVTime:INDependent:LENgth <value>
[SENSe]:PULSe:OGRAm[:MEASview<y>]:ANALysis:FVTime:INDependent:LENgth?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

<value> = <NRf>

Returns

<NRf> specifies the independent spectrum length in seconds.

Examples

SENSe:PULSe:OGRAm:MEASview1:ANALysis:FVTime:INDependent:LENgth? might return 10E-6s.

[SENSe]:PULSe[:MEASview<y>]:SIGNal:TYPE

Sets or queries the signal type in the pulsed RF measurements.

Conditions

Measurement views: Pulse Statistics with specified Measurement view number, Pulse table, Pulse trace with specified Measurement view number

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:SIGNal:TYPE { CWConst | CWCHange | LCHirp | OTHer}
[SENSe]:PULSe[:MEASview<y>]:SIGNal:TYPE?
```

Arguments

CWConst selects CW (continuous wave) with constant phase.

CWCHange selects CW (continuous wave) with changing phase.

LCHirp selects linear chirp.

OTHer selects other.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf>

Examples

SENSe:PULSe:MEASview2:SIGNal:TYPE CWCHange selects the CW with changing phase as the signal type.

[SENSe]:PULSe[:MEASview<y>]:STATistics:HISTogram:ORDinate

Sets or queries the Histogram Y unit. This command is valid when `DISPlay:PULSe:STATistics:PLOT` is set to Histogram.

Conditions

Measurement views: Pulse statistics

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:STATistics:HISTogram:ORDinate { PERCent | COUNt }
[SENSe]:PULSe[:MEASview<y>]:STATistics:HISTogram:ORDinate?
```

Arguments

`PERCent` sets the Histogram plot Y axis to the percentage of points.

`COUNt` sets the Histogram plot Y axis to the number of points in the bin.

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<NRf>`

Examples

`SENSe:PULSe:MEASview2:STATistics:HISTogram:ORDinate PERCENT` sets the histogram Y axis to percent.

[SENSe]:PULSe[:MEASview<y>]:STATistics:HISTogram:POINts

Sets or queries the number of bins in the Histogram plot.

Conditions

Measurement views: Pulse statistics

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:STATistics:HISTogram:POINts <value>
[SENSe]:PULSe[:MEASview<y>]:STATistics:HISTogram:POINts?
```

Arguments

`<value> ::= <NRf>` specifies the number of histogram bins.

Range: 5 to 10000

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<NRf>`

Examples

`SENSe:PULSe:MEASview2:STATistics:HISTogram:POINts 10` sets the number of histogram bins to 10.

[SENSe]:PULSe[:MEASview<y>]:STATistics:HISTogram:POINts:AUTO

Sets or queries enabling automatic Histogram plot bins count.

Conditions

Measurement views: Pulse statistics

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:STATistics:HISTogram:POINts:AUTO { OFF | ON | 0 | 1 }
[SENSe]:PULSe[:MEASview<y>]:STATistics:HISTogram:POINts:AUTO?
```

Arguments

`OFF` or `0` sets the number of bins manually by the `[SENSe]:PULSe:STATistics:HISTogram:POINts` command.

`ON` or `1` sets the number of bins to the number of pulses/10.

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<NRf>`

Examples

`SENSe:PULSe:MEASview2:STATistics:HISTogram:POINts:AUTO 1` sets the number of histogram bins based on number of pulses.

[SENSe]:PULSe[:MEASview<y>]:STATistics:IN:LINear:SCALe

Sets or queries the “Calculate statistics in linear scale” selection from the display.

Conditions

Measurement views: Pulse Cumulative Statistics, Pulse Statistics

Group

Sense commands

Syntax

```
[SENSe]:PULSe[:MEASview<y>]:STATistics:IN:LINear:SCALe { 0 | OFF | 1 | ON}
[SENSe]:PULSe[:MEASview<y>]:STATistics:IN:LINear:SCALe?
```

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

`0` or `OFF` unchecks the box and disables the calculate function

`1` or `ON` checks the box and enables the calculate function

Returns

<Boolean>, 0 or 1

Examples

SENSe:PULSe:MEASview1:STATistics:IN:LINear:SCALe 1 enables the Calculate statistics in linear scale (for results in log).

[SENSe]:PULSe:TRACe[:MEASview<y>]:SELEcted:SOURce

Sets or queries the Source of the selected Measurement view in pulse trace.

Conditions

Measurement views: Pulse trace with specified Measurement view number

Source cannot be changed when acquisition or analysis is in progress. Stop the acquisition and change the Source.

Group

Sense commands

Syntax

```
[SENSe]:PULSe:TRACe[:MEASview<y>]:SELEcted:SOURce {"Source 1" | "Source 2" | "Source 3" | "Source 4"}
[SENSe]:PULSe:TRACe[:MEASview<y>]:SELEcted:SOURce?
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf>

Examples

SENSe:PULSE:TRACE:MEASVIEW1:SELECTED:SOURCE? queries the source of the pulse trace measurement view 1.

[SENSe]:PULSe:RESult[:MEASview<y>]:SELEcted:SOURce

Sets or queries the Source of the selected Measurement view in pulse Result.

Conditions

Measurement views: Pulse Result with specified Measurement view number

Source cannot be changed when acquisition or analysis is in progress. Stop the acquisition and change the Source.

Group

Sense commands

Syntax

```
[SENSe]:PULSe:RESult[:MEASview<y>]:SELEcted:SOURce {"Source1"|"Source2"|"Source3"|"Source4"}
[SENSe]:PULSe:RESult[:MEASview<y>]:SELEcted:SOURce?
```

Arguments

<y>=1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1

Examples

[SENSe]:PULSe:RESult[:MEASview1:SELEcted:SOURce? Queries the Source of the spectrum Measurement View 1.

[SENSe]:REANalyze (No Query Form)

Sets all measurements to reanalyze the last acquisition.

NOTE: hg is an overlapped command, which does not finish executing before the next command starts executing. Use the *OPC(?) and *WAI commands to synchronize all pending operations to the execution of this command.

Conditions

Measurement views: All

Group

Sense commands

Syntax

[SENSe]:REANalyze

Related Commands

*OPC (on page 1544)

*WAI (on page 2739)

Arguments

None

Examples

SENSe:REANALYZE reanalyzes all measurements using the current acquisition.

[SENSe]:REANalyze:ACQuisition:SETTings? (Query Only)

Queries the settings of all acquisitions.

Conditions

Measurement views: All

Group

Sense commands

Syntax

[SENSe]:REANalyze:ACQuisition:SETTings?

[SENSe]:REANalyze:ALL:LOOP (No Query Form)

Replay all selected records/frames in a continuous loop.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:ALL:LOOP
```

[SENSe]:REANalyze:CURRent (No Query Form)

Replay the currently selected acquisition (or frame).

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:CURRent
```

[SENSe]:REANalyze:CURRent:ACQuisition? (Query Only)

Queries the Acquisition record identifier (ID) of the currently selected acquisition (or frame).

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:CURRent:ACQuisition?
```

[SENSe]:REANalyze:CURRent:FRAMe? (Query Only)

Queries the Frame identifier (ID) of the currently selected acquisition (or frame).

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:CURRent:FRAMe?
```

[SENSe]:REANalyze:CURRent (No Query Form)

Replay the currently selected acquisition (or frame).

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:CURRent
```

[SENSe]:REANalyze:CURRent:TIMestamp? (Query Only)

Queries the time stamp of the currently selected acquisition (or frame).

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:CURRent:TIMestamp?
```

[SENSe]:REANalyze:DELete:DATA (No Query Form)

Deletes all acquisition data.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:DELete:DATA
```

[SENSe]:REANalyze:FIRSt (No Query Form)

Replays (reanalyzes) the first item in the currently selected acquisition (or frame).

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:FIRSt
```

[SENSe]:REANalyze:LAST (No Query Form)

Replays (reanalyzes) the last item in the currently selected acquisition (or frame).

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:LAST
```

[SENSe]:REANalyze:NEXT (No Query Form)

Replays (reanalyzes) the next item in the currently selected acquisition (or frame).

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:NEXT
```

[SENSe]:REANalyze:PAUSe

Sets or queries the reanalysis paused state.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:PAUSe {OFF | ON | 0 | 1 }
[SENSe]:REANalyze:PAUSe?
```

Arguments

`ON` or `1` pauses (blocks) any acquisition until current reanalysis is completed.

`OFF` or `0` disables the reanalysis pause mode.

[SENSe]:REANalyze:PREVious (No Query Form)

Replays (reanalyzes) the previous item in the currently selected acquisition (or frame).

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:PREVious
```

[SENSe]:REANalyze:SELect:ACQuisition:DATA:TYPE

Sets or queries the acquisition data type.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:SELect:ACQuisition:DATA:TYPE {SPECtra | IQ }
[SENSe]:REANalyze:SELect:ACQuisition:DATA:TYPE?
```

Arguments

`SPECtra` sets the acquisition data type as DPX spectra data.

`IQ` sets the acquisition date type a IQ data.

Examples

`SENSe:REANALYZE:SELECT:ACQUISITION:DATA:TYPE IQ` sets the acquisition data type to IQ data.

[SENSe]:REANalyze:SELect:ACQuisition:FIRSt? (Query Only)

Queries the first selectable acquisition Start or Stop value in the currently selected set.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:SELect:ACQuisition:FIRSt?
```

Related Commands

[SENSe]:REANalyze:SELect:ACQuisition:FIRSt:TIMestamp? (on page 2311)

[SENSe]:REANalyze:SELect:ACQuisition:FIRSt:TIMestamp? (Query Only)

Queries the time stamp of the first selectable acquisition history Start or Stop value in the currently selected acquisition set.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:SELect:ACQuisition:FIRSt:TIMestamp?
```

Related Commands

[SENSe]:REANalyze:SELect:ACQuisition:FIRSt? (on page 2311)

[SENSe]:REANalyze:SELect:ACQuisition:LAST? (Query Only)

Queries the last selectable acquisition Start or Stop value in the currently selected set.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:SELect:ACQuisition:LAST?
```

Related Commands

[SENSe]:REANalyze:SELect:ACQuisition:LAST:TIMestamp? (on page 2312)

[SENSe]:REANalyze:SELect:ACQuisition:LAST:TIMestamp? (Query Only)

Queries the time stamp of the last selectable acquisition Start or Stop value in the currently selected set.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:SELect:ACQuisition:LAST:TIMestamp?
```

Related Commands

[SENSe]:REANalyze:SELect:ACQuisition:LAST? (on page 2312)

[SENSe]:REANalyze:SELect:ACQuisition:STARt

Sets or queries the Acquisition Start value in the currently selected set.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:SELect:ACQuisition:STARt <value>
[SENSe]:REANalyze:SELect:ACQuisition:STARt?
```

Arguments

`<value>::<NR1>` is an integer in the range from one to the number of acquisitions in the acquisition history.

[SENSe]:REANalyze:SELect:ACQuisition:STARt:TIMestamp? (Query only)

Queries the time stamp of the current Acquisition Start selection. It is the time stamp of the first frame of the selected acquisition, if FastFrame is enabled.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:SELect:ACQuisition:STARt:TIMestamp?
```

[SENSe]:REANalyze:SELect:ACQuisition:STOP

Sets or queries the Acquisition Stop value in the currently selected set.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:SELect:ACQuisition:STOP <value>
[SENSe]:REANalyze:SELect:ACQuisition:STOP?
```

Arguments

`<value>::<NR1>` is an integer in the range from one to the number of acquisitions in the acquisition history.

[SENSe]:REANalyze:SELect:ACQuisition:STOP:TIMestamp? (Query Only)

Queries the time stamp of the current Acquisition Stop selection. This is the time stamp of the last frame of the selected acquisition, if FastFrame is enabled.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:SELect:ACQuisition:STOP:TIMestamp?
```

[SENSe]:REANalyze:SELect:ALL (No Query Form)

Selects all acquisitions and frames in the currently selected set.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:SELect:ALL
```

[SENSe]:REANalyze:SELect:FRAMe:FIRSt? (Query Only)

Queries the first available Frame Start or Stop value available in the currently selected set.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:SELect:FRAMe:FIRSt?
```

Related Commands

[SENSe]:REANalyze:SELect:FRAMe:FIRSt:TIMestamp? (on page 2314)

[SENSe]:REANalyze:SELect:FRAMe:FIRSt:TIMestamp? (Query Only)

Queries the time stamp of the first available frame of the first available Acquisition Start or Stop value available in the currently selected set.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:SELect:FRAMe:FIRSt:TIMestamp?
```

Related Commands

[SENSe]:REANalyze:SELect:FRAMe:FIRSt? (on page 2314)

[SENSe]:REANalyze:SELect:FRAMe:LAST? (Query Only)

Queries the last available Frame Start or Stop value available in the currently selected set.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:SELect:FRAMe:LAST?
```

Related Commands

[SENSe]:REANalyze:SELect:FRAMe:LAST:TIMestamp? (on page 2315)

[SENSe]:REANalyze:SELect:FRAMe:LAST:TIMestamp? (Query Only)

Queries the time stamp of the last available frame of the last available Acquisition Start or Stop value available in the currently selected set.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:SELect:FRAMe:LAST:TIMestamp?
```

Related Commands

[SENSe]:REANalyze:SELect:FRAMe:LAST? (on page 2315)

[SENSe]:REANalyze:SELect:FRAMe:STARt

Sets or queries the Frame Start value in the currently selected set.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:SELect:FRAMe:STARt <value>
[SENSe]:REANalyze:SELect:FRAMe:STARt?
```

Arguments

`<value>::<NR1>` is an integer in the range from one to the number of fast frames.

[SENSe]:REANalyze:SELect:FRAMe:STARt:TIMestamp? (Query Only)

Queries the time stamp of the current Frame Start selection. This is the time stamp of the Frame Start frame of the acquisition start acquisition, if FastFrame is enabled.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:SELect:FRAMe:STARt:TIMestamp?
```

[SENSe]:REANalyze:SELect:FRAMe:STOP

Sets or queries the frame stop value in the currently selected set.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:SELect:FRAMe:STOP <value>
[SENSe]:REANalyze:SELect:FRAMe:STOP?
```

Related Commands

[SENSe]:REANalyze:SELect:FRAMe:STOP:TIMestamp? (on page 2316)

Arguments

`<value>::<NR1>` is an integer in the range from one to the number of fast frames.

[SENSe]:REANalyze:SELect:FRAMe:STOP:TIMestamp? (Query Only)

Queries the time stamp of the current frame stop selection. It is the time stamp of the frame Stop frame of the Acquisition Stop acquisition if FastFrame is enabled.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:SELect:FRAMe:STOP:TIMestamp?
```

Related Commands

[SENSe]:REANalyze:SELect:FRAMe:STOP (on page 2316)

[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:ALL (No Query Form)

Selects all spectra acquisitions and frames in the currently selected set.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:ALL
```

[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:CURRent:ACQuisition? (Query Only)

Queries the acquisition record identifier (ID) of the currently selected acquisition.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:CURRent:ACQuisition?
```

Returns

`<ID Number> ::= <NR1>` the record identifier number of the current acquisition.

Examples

`SENSe:REANALYZE:SELECT:SPECTRA:ACQUISITION:CURRENT:ACQUISITION?` might return 848 indicating that the current acquisition record number is 848.

[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:CURRent:TIMestamp? (Query Only)

Queries the time stamp of the currently selected acquisition.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:CURRent: TIMestamp?
```

Returns

<time stamp> ::= <string> The current time stamp value.

Examples

SENSe:REANALYZE:SELECT:SPECTRA:ACQUISITION:CURRENT:TIMESTAMP? might return “10/10/2011 10:57:55:93” indicating that the current time stamp recorded on October 10, 2011 was 10 hours, 57 minutes, and 55.93 seconds.

[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:FIRSt? (Query Only)

Queries the first selectable acquisition start or stop value in the currently selected set.

Conditions

Measurement views: All

Group

Sense commands

Syntax

[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:FIRSt?

Returns

<ID Number> ::= <NR1> the record identifier number of the current acquisition.

Examples

SENSe:REANALYZE:SELECT:SPECTRA:ACQUISITION:FIRST? might return 1 indicating that the first selectable acquisition start or stop value was 1.

[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:FIRSt:TIMestamp? (Query Only)

Queries the time stamp of the first selectable acquisition history start or stop value in the currently selected acquisition set.

Conditions

Measurement views: All

Group

Sense commands

Syntax
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:FIRSt:TIMestamp?

Returns

<time stamp> ::= <string> The first time stamp value.

Examples

SENSe:REANALYZE:SELECT:SPECTRA:ACQUISITION:FIRST:TIMESTAMP? might return "10/10/2011 10:57:41:58", indicating that the first time stamp recorded on October 10, 2011 was 10 hours, 57 minutes, and 41.58 seconds.

[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:LAST? (Query Only)

Queries the last selectable acquisition start or stop value in the currently selected set.

Conditions

Measurement views: All

Group

Sense commands

Syntax

`[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:LAST?`

Returns

`<ID Number> ::= <NR1>` the record identifier number of the current acquisition.

Examples

`SENSe:REANALYZE:SELECT:SPECTRA:ACQUISITION:LAST?` might return `1`, indicating that the last selectable acquisition start or stop value was 1.

[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:LAST:TIMestamp? (Query Only)

Queries the time stamp of the last selectable acquisition start or stop value in the currently selected set.

Conditions

Measurement views: All

Group

Sense commands

Syntax

`[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:LAST:TIMestamp?`

Returns

`<time stamp> ::= <string>` The last time stamp value.

Examples

`SENSe:REANALYZE:SELECT:SPECTRA:ACQUISITION:LAST:TIMESTAMP?` might return `"10/10/2011 10:57:41:58"` indicating that the last time stamp recorded on October 10, 2011 was 10 hours, 57 minutes, and 41.58 seconds.

[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:STARt

Sets or queries the acquisition start value in the currently selected set.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:STARt <value>
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:STARt?
```

Arguments

`<value> ::= <NR1>` is an integer in the range from one to the number of acquisitions in the acquisition history.

Examples

`SENSe:REANALYZE:SELECT:SPECTRA:ACQUISITION:START?` might return `1.0000000000`, indicating the start value was 1.

[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:STARt:TIMestamp? (Query Only)

Queries the time stamp of the current acquisition start selection.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:STARt:TIMestamp?
```

Returns

`<time stamp> ::= <string>` The start time stamp value.

Examples

`SENSe:REANALYZE:SELECT:SPECTRA:ACQUISITION:START:TIMESTAMP?` might return `"10/10/2011 10:57:41:58"` indicating that the time stamp of the acquisition start selection recorded on October 10, 2011 was 10 hours, 57 minutes, and 41.58 seconds.

[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:STOP

Sets or queries the acquisition stop value in the currently selected set.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:STOP <value>
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:STOP?
```

Arguments

<value> ::= <NR1> is an integer in the range from one to the number of acquisitions in the acquisition history.

Examples

SENSe:REANALYZE:SELECT:SPECTRA:ACQUISITION:STOP? might return 1.0000000000, indicating the stop value was 1.

[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:STOP:TIMestamp? (Query Only)

Queries the time stamp of the current acquisition stop selection.

Conditions

Measurement views: All

Group

Sense commands

Syntax

[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:STOP:TIMestamp?

Returns

<time stamp> ::= <string> The stop time stamp value.

Examples

SENSe:REANALYZE:SELECT:SPECTRA:ACQUISITION:STOP:TIMESTAMP? might return "10/10/2011 10:57:41:58" indicating that the time stamp of stop selection recorded on October 10, 2011 was 10 hours, 57 minutes, and 41.58 seconds.

[SENSe]:REANalyze:SPEed

Sets or queries the time delay value inserted between reanalysis of frames or acquisitions.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:SPEed <value>
[SENSe]:REANalyze:SPEed?
```

Arguments

<value>::<NR1> is an integer in the range of zero (for no delay) to four (for a maximum delay of about one second).

[SENSe]:REANalyze:STOP (No Query Form)

Stops the current reanalysis. The command will block until the current reanalysis has completed and stopped.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:STOP
```

[SENSe]:REANalyze:TIMestamp:DECimal

Sets or queries the number of values to the right of the decimal point to include in all time stamps.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:REANalyze:TIMestamp:DECimal <value>
[SENSe]:REANalyze:TIMestamp:DECimal?
```

Arguments

`<value>::<NR1>` is an integer in the range of one to nine.

SENSe:ROSCillator:EXTernal:TIMe

Sets or queries the timing reference.

If you set the source to GPS, PPS, or IRIG-B after launch of SignalVu-PC (thus changing the time from the default system time), and then disconnect from that source or otherwise lose the lock to that source, the time used will be that of the previous externally generated source. This will remain so until you set a different source with a valid signal.

Group

Sense commands

Syntax

```
SENSe:ROSCillator:EXTernal:TIMe {PPS | IBAM | IBDC | GPS | SYStem}
SENSe:ROSCillator:EXTernal:TIMe?
```

Related Commands

SENSe:ROSCillator:EXTernal:TIMe:STATus? (on page 2323)

Arguments

`PPS` is a 1PPS signal

`IBAM` is an IRIG-B AM signal

`IBDC` is an IRIG-B DC signal

`GPS` is a GNSS signal

`SYStem` is the CTRL7100A (PC) time

Returns

One of the following: PPS, IBAM, IBDC, GPS, SYStem

Examples

`SENSe:ROSCILLATOR:EXTERNAL:TIME PPS` sets the system time to 1 PPS.

SENSe:ROSCillator:EXTernal:TIMe:STATus? (Query Only)

Queries if the timing reference source is locked or unlocked.

Group

Sense commands

Syntax

```
SENSe:ROSCillator:EXTernal:TIMe:STATus?
```

Related Commands

SENSe:ROSCillator:EXTernal:TIMe (on page 2322)

Returns

0 means the timing source is unlocked.

1 means the timing source is locked.

Examples

`SENSe:ROSCILLATOR:EXTERNAL:TIME:STATUS?` might return 1, indicating the timing source is locked. This is indicated in the UI on the Timing Reference tab by a green checked box next to Valid Timing Source.

[SENSe]:ROSCillator:SOURce

Sets or queries the frequency reference oscillator source.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:ROSCillator:SOURce { INTernal | EXTernal }
[SENSe]:ROSCillator:SOURce?
```

Related Commands

SENSe:ROSCillator:EXTernal:TIMe:STATus? (on page 2323)

Arguments

`INTernal` specifies that the analyzer uses the 10 MHz internal oscillator.

`EXTernal` specifies that the analyzer uses the external reference signal. Use the Ref In connector on the rear panel to input the signal.

Selecting `EXTernal` initiates an attempt to lock the internal reference oscillator to the external reference signal. If the signal is not connected or is at an invalid frequency or amplitude, an error (2028, "External frequency reference signal not valid. Using internal reference) is returned. If not able to lock to the external reference, an error (2029, "Unable to lock to external frequency reference. Using internal reference.") is returned.

Examples

`SENSe:ROSCILLATOR:SOURCE INTernal` uses the 10 MHz internal oscillator for the frequency reference.

[SENSe]:SEM:{BANDwidth|BWIDth}[:RESolution]

Sets or queries the resolution bandwidth (RBW) of the reference channel.

NOTE: In Real-time operation this sets the RBW for the reference and all offset channels.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:{BANDwidth|BWIDth}[:RESolution] <value>
[SENSe]:SEM:{BANDwidth|BWIDth}[:RESolution]?
```

Arguments

`<value>::<NRf>` specifies the resolution bandwidth.

Examples

`SENSe:SEM:BANDWIDTH:RESOLUTION 200kHZ` sets the resolution bandwidth to 200 kHz.

[SENSe]:SEM:{BANDwidth|BWIDth}:VIDeo

Sets or queries the video bandwidth (VBW) of the reference channel.

NOTE: In Real-time operation this sets the RBW for the reference and all offset channels.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:{BANDwidth|BWIDth}:VIDeo <value>
[SENSe]:SEM:{BANDwidth|BWIDth}:VIDeo?
```

Arguments

`<value>::<NRf>` specifies the video bandwidth.

Range: Current RBW/104 (1 Hz minimum) to current RBW.

Examples

`SENSe:SEM:{BANDWIDTH|BWIDTH}:VIDEO 200kHZ` sets the video bandwidth to 200 kHz.

[SENSe]:SEM:{BANDwidth|BWIDth}:VIDeo:STATe

Sets or queries the state of the video bandwidth (VBW) of the reference channel.

NOTE: In Real-time operation this sets the VBW state for the reference and all offset channels.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:{BANDwidth|BWIDth}:VIDeo:STATe { OFF | ON | 0 | 1 }
[SENSe]:SEM:{BANDwidth|BWIDth}:VIDeo:STATe?
```

Arguments

`OFF` or `0` turns the video bandwidth filter off.

`ON` or `1` turns the video bandwidth filter on.

[SENSe]:SEM:CHANnel:FILTer

Sets or queries the reference Channel Filter.

NOTE: In Real-time operation the channel filter is always forced to None.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:CHANnel:FILTer { RRC | NONE }
[SENSe]:SEM:CHANnel:FILTer?
```

Arguments

`RRC` selects the Root-Raised-Cosine filter.

`NONE` uses no filter.

Examples

`SENSe:SEM:CHANNEL:FILTER RRCosine` selects Root-Raised-Cosine for the reference Channel Filter.

[SENSe]:SEM:CHANnel:IBANdwidth

Sets or queries the reference channel integration bandwidth.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:CHANnel:IBANdwidth <value>
[SENSe]:SEM:CHANnel:IBANdwidth?
```

Arguments

`<value>::<NRf>` integration bandwidth

Returns

Examples

`SENSe:SEM:CHANNEL:IBANDWIDTH?` might return `18.000000E+6`, indicating that the integration bandwidth is 18 MHz.

[SENSe]:SEM:CHANnel:WIDTh

Sets or queries the reference Channel bandwidth.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:CHANnel:WIDTh <value>
[SENSe]:SEM:CHANnel:WIDTh?
```

Arguments

`<value>::<NRf>` specifies the filter bandwidth.

Examples

`SENSe:SEM:CHANNEL:WIDTH 30kHZ` sets the filter bandwidth to 30 kHz.

[SENSe]:SEM:CHIPrate

Sets or queries the Chip rate.

Use the `[SENSe]:SEM:CHANnel:FILTer` command to set the reference Channel Filter to RRC (Root-Raised Cosine).

NOTE: Applies in Swept operation only.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:CHIPrate <value>
[SENSe]:SEM:CHIPrate?
```

Arguments

`<value>::<NRf>` specifies the chip rate.

Examples

`SENSe:SEM:CHIPRATE 3.84MHZ` sets the Chip rate toe 3.84 MHz.

[SENSe]:SEM:CLEar:RESults (No Query Form)

Clears the results in the Spectral Emissions Mask measurement.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:CLEar:RESults
```

Arguments

None

Examples

`SENSe:SEM:CLEAR:RESULTS` clears the results in the measurement.

[SENSe]:SEM:FREQuency

Sets or queries the measurement frequency.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:FREQuency <value>
[SENSe]:SEM:FREQuency?
```

Arguments

`<value>::<NRf>` specifies the center frequency.

Examples

`[SENSE]:SEM:FREQUENCY 1.0GHZ` sets the center frequency to 1.0 GHz.

[SENSe]:SEM:FREQuency:SPAN? (Query Only)

Returns the frequency span.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:FREQuency:SPAN?
```

Arguments

None

Returns

<NRf> frequency span.

Examples

SENSe:SEM:FREQUENCY:SPAN? might return 100.00000E+6, indicating that the frequency span is 100 MHz.

[SENSe]:SEM:MFILter

Sets or queries the shape of the filter.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:MFILter { GAUSsian | RECTangular }
[SENSe]:SEM:MFILter?
```

Arguments

GAUSsian sets the filter shape to Gaussian.

RECTangular sets the filter shape to Rectangular.

Examples

SENSe:SEM:MFILTER RECTangular sets the filter shape to Rectangular.

[SENSe]:SEM:MODE:REALtime

Sets or queries the Real-time SEM measurement mode.

NOTE: In real-time mode SEM is measured using real-time acquisition data.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:MODE:REALtime { OFF | ON | 0 | 1 }
[SENSe]:SEM:MODE:REALtime?
```

Arguments

`ON` or `1` turns the SEM real-time measurement mode on.

`OFF` or `0` turns the SEM real-time measurement mode off.

[SENSe]:SEM:MTYPe

Sets or queries the measurement type.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:MTYPe { TPOW | PSD | PEAK }
```

Arguments

`TPOW` specifies the Total Power measurement.

`PSD` specifies the Peak Spectral Density measurement in dBM/Hz.

`PEAK` specifies a Peak power measurement.

Examples

`SENSe:SEM:MTYPE PEAK` sets the measurement type to Peak.

[SENSe]:SEM:OFFSet:DETection

Sets or queries the detection settings for all offsets.

NOTE: In Real-time operation the offset and reference detection settings are forced equal.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:OFFSet:DETection { AVERage | PEAK }
[SENSe]:SEM:OFFSet:DETection?
```

Arguments

`AVERage` indicates the Offsets setting is Average (VRMS).

`PEAK` indicates the Offsets setting is the positive peak.

Examples

`SENSe:SEM:OFFSET:DETECTION AVER` sets the detection settings for all offsets to Average RMS.

[SENSe]:SEM:OFFSet<x>:BANDwidth:FILTer? (Query Only)

Returns the bandwidth filter setting for specified row in the Offsets & Limits table. The filter setting is equal to the Resolution Bandwidth (RBW) value times the number of resolution bandwidths. The parameter <x> represents the row in the Offsets & Limits table.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:OFFSet<x>:BANDwidth:FILTer?
```

Arguments

None

Returns

`<value>::<NRf>` is the filter bandwidth setting.

Examples

`SENSe:SEM:OFFSet3:BANDwidth:FILTer` might return `30.0E+3`, indicating that the filter bandwidth for row 3 to 30 kHz.

[SENSe]:SEM:OFFSet<x>:BANDwidth[:RESolution]

Sets or queries the resolution bandwidth for specified row in the Offsets & Limits table. The parameter <x> represents the row in the Offsets & Limits table. See the [SENSe]:SEM:{BANDwidth|BWIDth}[:RESolution] (on page 2324) command.

NOTE: In Real-time operation the RBW values for each offset and the reference are set equal to the All Channels RBW setting.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:OFFSet<x>:BANDwidth[:RESolution] <value>
[SENSe]:SEM:OFFSet<x>:BANDwidth[:RESolution]?
```

Returns

<value>::<NRf> specifies the resolution bandwidth.

Examples

SENSe:SEM:OFFSet4:BANDwidth 30kHz sets the resolution bandwidth in row 4 to 30 kHz.

[SENSe]:SEM:OFFSet<x>:BANDwidth:RESolution:COUNt

Sets or queries the count of Resolution Bandwidths (RBWs) for specified row in the Offsets & Limits table. The parameter <x> represents the row in the Offsets & Limits table.

NOTE: In Real-time operation the RBW count is always forced to 1.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:OFFSet<x>:BANDwidth:RESolution:COUNt <value>
[SENSe]:SEM:OFFSet<x>:BANDwidth:RESolution:COUNt?
```

Arguments

<value> specifies the count number of RBWs.

Examples

SENSe:SEM:OFFSet1:BANDwidth:RESolution:COUNt 10 sets the count for row 1 to 10.

[SENSe]:SEM:OFFSet<x>:BANDwidth:VIDeo

Sets or queries the video bandwidth setting for specified row in the Offsets & Limits table. The parameter <x> represents the row in the Offsets & Limits table. See the [SENSe]:SEM:{BANDwidth|BWIDth}:VIDeo (on page 2324) command.

NOTE: In Real-time operation the VBW values for each offset and the reference are set equal to the All Channels VBW setting.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:OFFSet<x>:BANDwidth:VIDeo <value>
[SENSe]:SEM:OFFSet<x>:BANDwidth:VIDeo?
```

Arguments

<value>::<NRf> specifies the video bandwidth setting.

Examples

`SENSe:SEM:OFFSet3:BANDwidth:VIDeo 300` sets the video bandwidth for row 3 to 300 Hz.

[SENSe]:SEM:OFFSet<x>:BANDwidth:VIDeo:STATe

Sets or queries the state of the video bandwidth for specified row in the Offsets & Limits table. The parameter <x> represents the row in the Offsets & Limits table. See the [SENSe]:SEM:{BANDwidth|BWIDth}:VIDeo:STATe (on page 2325) command.

NOTE: In Real-time operation the VBW state values for each offset and the reference are set equal to the All Channels VBW state setting.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:OFFSet<x>:BANDwidth:VIDeo:STATe { OFF | ON | 0 | 1 }
[SENSe]:SEM:OFFSet<x>:BANDwidth:VIDeo:STATe?
```

Arguments

`OFF` or `0` turns the video bandwidth filter off.

`ON` or `1` turns the video bandwidth filter on.

Examples

`SENSe:SEM:OFFSet3:BANDwidth:VIDeo:STATe ON` turns on the video bandwidth filter for row 3 in the table.

[SENSe]:SEM:OFFSet<x>:FREQuency:STARt

Sets or queries the start frequency for specified row in the Offsets & Limits table. The parameter <x> represents the row in the Offsets & Limits table.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:OFFSet<x>:FREQuency:STARt <value>
[SENSe]:SEM:OFFSet<x>:FREQuency:STARt?
```

Arguments

`<value>::<NRf>` specifies the start frequency.

Examples

`SENSe:SEM:OFFSet1:FREQuency:STARt 2.876MHZ` sets the start frequency in row 1 to 2.876 MHz.

[SENSe]:SEM:OFFSet<x>:FREQuency:STOP

Sets or queries the stop frequency for specified row in the Offsets & Limits table. The parameter <x> represents the row in the Offsets & Limits table.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:OFFSet<x>:FREQuency:STOP <value>
[SENSe]:SEM:OFFSet<x>:FREQuency:STOP?
```

Arguments

`<value>::<NRf>` specifies the stop frequency.

Examples

`SENSe:SEM:OFFSet1:FREQuency:STOP 3.485MHZ` sets the stop frequency in row 1 to 3.485 MHz.

[SENSe]:SEM:OFFSet<x>:LIMIT:ABSolute:SAME

Sets or queries the state of the limits for the absolute start and stop settings for specified row in the Offsets & Limits table. The parameter <x> represents the row in the Offsets & Limits table.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:OFFSet<x>:LIMIT:ABSolute:SAME { OFF | ON | 0 | 1 }
[SENSe]:SEM:OFFSet<x>:LIMIT:ABSolute:SAME?
```

Arguments

`OFF` or `0` indicates the start and stop limits are different.

`ON` or `1` indicates the state and stop limits are the same.

Examples

`SENSe:SEM:OFFSet2:LIMIT:ABSolute:SAME ON` sets the absolute start and stop limits for row 2 to the same values.

[SENSe]:SEM:OFFSet<x>:LIMIT:ABSolute:STARt

Sets or queries the absolute power start limit for specified row in the Offsets & Limits table. The parameter <x> represents the row in the Offsets & Limits table.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:OFFSet<x>:LIMIT:ABSolute:STARt <value>
[SENSe]:SEM:OFFSet<x>:LIMIT:ABSolute:STARt?
```

Arguments

`<value>::<NRf>` specifies the power in dBm.

Examples

`SENSe:SEM:OFFSet1:LIMIT:ABSolute:STARt -71.10` sets the absolute start power limit for row 1 to −71.10 dBm.

[SENSe]:SEM:OFFSet<x>:LIMIT:ABSolute:STOP

Sets or queries the absolute power stop limit for specified row in the Offsets & Limits table. The parameter <x> represents the row in the Offsets & Limits table

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:OFFSet<x>:LIMIT:ABSolute:STOP <value>
[SENSe]:SEM:OFFSet<x>:LIMIT:ABSolute:STOP?
```

Arguments

`<value>::<NRf>` specifies the power in dBm.

Examples

`SENSe:SEM:OFFSet<x>:LIMIT:ABSolute:STOP -55.80` sets the absolute stop power limit for row 1 to −55.80 dBm.

[SENSe]:SEM:OFFSet<x>:LIMIT:MASK

Sets or queries the limit mask setting for the specified row in the Offsets & Limits table. The parameter <x> represents the row in the Offsets & Limits table.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:OFFSet<x>:LIMIT:MASK { ABS | REL | AND | OR }
[SENSe]:SEM:OFFSet<x>:LIMIT:MASK?
```

Arguments

`ABS` specifies the limit mask is set to Absolute.

REL specifies the limit mask is set to Relative

AND specifies the limit mask is set to Absolute AND Relative.

OR specifies the limit mask is set to Absolute OR Relative.

Examples

SENSe:SEM:OFFSet3:LIMIT:MASK REL sets the limit mask for row 3 to Relative.

[SENSe]:SEM:OFFSet<x>:LIMIT:RELative:SAME

Sets or queries the state of the limits for the relative start and stop settings for specified row in the Offsets & Limits table. The parameter <x> represents the row in the Offsets & Limits table.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:OFFSet<x>:LIMIT:RELative:SAME { OFF | ON | 0 | 1 }
[SENSe]:SEM:OFFSet<x>:LIMIT:RELative:SAME?
```

Arguments

OFF or 0 indicates the start and stop limits are different.

ON or 1 indicates the state and stop limits are the same.

Examples

SENSe:SEM:OFFSet2:LIMIT:RELative:SAME ON sets the relative start and stop limits for row 2 to the same values.

[SENSe]:SEM:OFFSet<x>:LIMIT:RELative:STARt

Sets or queries the relative power start limit for specified row in the Offsets & Limits table. The parameter <x> represents the row in the Offsets & Limits table.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:OFFSet<x>:LIMIT:RELative:STARt <value>
[SENSe]:SEM:OFFSet<x>:LIMIT:RELative:STARt?
```

Arguments

<value>::<NRf> specifies the power in dB.

Examples

SENSe:SEM:OFFSet1:LIMIT:RELative:STARt -35.5 sets the relative start power limit for row 1 to −35.5 dB.

[SENSe]:SEM:OFFSet<x>:LIMIT:RELative:STOP

Sets or queries the relative power stop limit for specified row in the Offsets & Limits table. The parameter <x> represents the row in the Offsets & Limits table.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:OFFSet<x>:LIMIT:RELative:STOP <value>
[SENSe]:SEM:OFFSet<x>:LIMIT:RELative:STOP?
```

Arguments

`<value>::<NRf>` specifies the power in dB.

Examples

`SENSe:SEM:OFFSet1:LIMIT:RELative:STOP -35.5` sets the relative stop power limit for row 1 to –35.5 dB.

[SENSe]:SEM:OFFSet<x>:LIMIT:SIDE

Sets or queries the side of the offset limit for specified row in the Offsets & Limits table. The parameter <x> represents the row in the Offsets & Limits table.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:OFFSet<x>:LIMIT:SIDE { LEFT | RIGHt | BOTH}
[SENSe]:SEM:OFFSet<x>:LIMIT:SIDE?
```

Arguments

`LEFT` specifies the left offset limit.

`RIGHt` specifies the right offset limit.

`BOTH` specifies both sides of the offset limit.

Examples

`SENSe:SEM:OFFSet1:LIMIT:SIDE BOTH` specifies the measurement uses both sides of the offset in row 1.

[SENSe]:SEM:OFFSet<x>:STATe (No Query Form)

Determines whether to display the specified row of the Offsets & Limits table. The parameter <x> represents the row in the Offsets & Limits table.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:OFFSet<x>:STATe { OFF | ON | 0 | 1 }
```

Arguments

`OFF` or `0` turns off the display of the row in the measurement view.

`ON` or `1` displays the row in the measurement view.

Examples

`SENSe:SEM:OFFSet2:STATe OFF` the display of row 2 in the measurement view.

[SENSe]:SEM:POINts:COUNt

Sets or queries the maximum number of trace points per range.

NOTE: In Swept operation this is the number of trace points per range. In Real-time operation it is the number of trace points in the entire trace.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:POINts:COUNt <value>
[SENSe]:SEM:POINts:COUNt?
```

Arguments

None

Returns

`P<NR1>` number of points per range.

Examples

`SENSe:SEM:POINTS:COUNT?` might return `P8001`, indicating a maximum of 8001 trace points.

[SENSe]:SEM:PRESet (No Query Form)

Defines the standard protocol settings for the Spectral Emissions Mask measurement.

NOTE: This is an obsolete command.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:PRESet { "3GPP Downlink" | "WLAN 802.11g" | "WiMax 802.16 (ETSI)" }
```

Arguments

"`3GPP Downlink`" specifies the 3GPP Downlink standard.

"`WLAN 802.11g`" specifies the IEEE 802.11 standard for WLAN communications.

"`WiMax 802.16 (ETSI)`" specifies the WiMAX 802.16 (ETSI) standard. Quotes must be included with the arguments.

Examples

`SENSe:SEM:PRESET WLAN (802.11g)` presets the SEM measurement to the IEEE 802.11 standard for WLAN communications.

[SENSe]:SEM:REFerence:AUTO

Determines whether to automatically update the Power Reference in the Spectral Emissions Mask measurement.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:REFerence:AUTO { OFF | ON | 0 | 1 }
[SENSe]:SEM:REFerence:AUTO?
```

Arguments

`OFF` or `0` turns off the automatic update.

`ON` or `1` automatically updates the reference power.

Examples

`SENSe:SEM:REFERENCE:AUTO ON` automatically updates the reference power in the measurement.

[SENSe]:SEM:REFerence:DETection

Sets or returns the reference channel detection setting.

NOTE: In Real-time operation the offset and reference detection settings are forced equal.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:REFerence:DETection { AVERage | PEAK }
[SENSe]:SEM:REFerence:DETection?
```

Arguments

`AVERage` indicates the reference channel detection setting is Average (VRMS).

`PEAK` indicates the reference channel detection setting is the positive peak.

Examples

`SENSe:SEM:REFERENCE:DETECTION AVER` sets the reference channel detection setting to Average RMS.

[SENSe]:SEM:REFerence[:MAGNitude]

Sets or queries the reference power magnitude.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:REFerence[:MAGNitude] <value>
[SENSe]:SEM:REFerence[:MAGNitude]?
```

Arguments

`<value>::<NRf>` specifies the magnitude in dBm.

Examples

`SENSe:SEM:REFERENCE[:MAGNITUDE]?` might return a magnitude of `-51.057`.

[SENSe]:SEM:RRCRolloff

Sets or queries the filter parameter (roll-off ratio) for the Root Raised Cosine filter.

NOTE: Applies in Swept operation only.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:RRCRolloff <value>
[SENSe]:SEM:RRCRolloff?
```

Arguments

`<value>::<NRf>` specifies the filter parameter.

Range: 0.001 to 1, 0.001 step.

Examples

`SENSe:SEM:RRCROLLOFF 0.22` sets the filter parameter to 0.22.

[SENSe]:SEM:STEP:AUTO

Determines whether to automatically update the center frequency increment/decrement step size in the Spectral Emissions Mask measurement. The Step value controls the magnitude of each step when using the front panel controls or arrow keys on the keyboard.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:STEP:AUTO { OFF | ON | 0 | 1 }
[SENSe]:SEM:STEP:AUTO?
```

Related Commands

[SENSe]:SEM:STEP[:MAGNitude] (on page 2340)

Arguments

`OFF` or 0 turns off the automatic update.

`ON` or 1 automatically updates center frequency.

Examples

`SENSe:SEM:STEP:AUTO ON` automatically updates the center frequency.

[SENSe]:SEM:STEP[:MAGNitude]

Sets or queries the step magnitude for the center frequency in the Spectral Emissions Mask measurement.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:STEP[:MAGNitude] <value>
[SENSe]:SEM:STEP[:MAGNitude]?
```

Related Commands

[SENSe]:SEM:STEP:AUTO (on page 2340)

Arguments

`<value>::<NRf>` specifies the step setting.

Examples

SENSe:SEM:STEP[:MAGNITUDE] 4MHZ sets the step magnitude to 4 MHz.

[SENSe]:SEM:WIDTh:DEFinition

Sets or queries the offset definition of the measurement type in the Spectral Emissions Mask measurement.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:WIDTh:DEFinition {C2C | C2E | E2C | E2E}
[SENSe]:SEM:WIDTh:DEFinition?
```

Arguments

C2C defines the offset from the reference frequency center to the offset center.

C2E defines the offset from the reference frequency center to the offset edge.

E2C defines the offset from the reference frequency edge to the offset center.

E2E defines the offset from the reference frequency edge to the offset edge.

Examples

SENSe:SEM:WIDTH:DEFINITION C2C specifies the offset limit from the center of the reference frequency to the center of the offset.

[SENSe]:SGRam[:MEASview<y>]:{BANDwidth|BWIDth}[:RESolution]

Sets or queries the resolution bandwidth (RBW). Programming a specified RBW sets SENSe:SGRam[:MEASview<y>]:{BANDwidth|BWIDth}[:RESolution]:AUTO OFF.

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:{BANDwidth|BWIDth}[:RESolution] <value>
[SENSe]:SGRam[:MEASview<y>]:{BANDwidth|BWIDth}[:RESolution]?
```

Related Commands

[SENSe]:SGRam[:MEASview<y>]:{BANDwidth|BWIDth}:RESolution:AUTO (on page 2342)

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

<value>::<NRf> specifies the RBW. Range: 1 Hz to 10 MHz.

Returns

<NRf> specifies the RBW.

Examples

`[SENSe]:SGRam:MEASview1:BANDwidth:RESolution 200kHz` sets the RBW to 200 kHz.

[SENSe]:SGRam[:MEASview<y>]:{BANDwidth|BWIDth}[:RESolution]:ACTual (Query Only)

Queries the actual resolution bandwidth (RBW) in the spectrogram.

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:{BANDwidth|BWIDth}[:RESolution]:ACTual?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Returns

<NRf> The actual RBW in Hz.

Examples

`SENSe:SGRam:MEASview1:BANDwidth:RESolution:ACTual?` might return `299.624E+3`, indicating that the actual RBW is 299.624 kHz.

[SENSe]:SGRam[:MEASview<y>]:{BANDwidth|BWIDth}[:RESolution]:AUTO

Determines whether to set the resolution bandwidth (RBW) automatically or manually in the spectrogram.

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:{BANDwidth|BWIDth}[:RESolution]:AUTO { OFF | ON | 0 | 1 }
[SENSe]:SGRam[:MEASview<y>]:{BANDwidth|BWIDth}[:RESolution]:AUTO?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

OFF or 0 specifies that the resolution bandwidth is set manually using the [SENSe]:DPX:{BANDwidth|BWIDth}:ACTual? command.

ON or 1 specifies that the resolution bandwidth is set automatically.

Returns

{ OFF | ON | 0 | 1 }

Examples

SENSe:SGRam:MEASview1:BANDwidth:RESolution:AUTO ON sets the resolution bandwidth automatically.

[SENSe]:SGRam[:MEASview<y>]:{BANDwidth|BWIDth}[:RESolution]:MODE

Determines whether to enable or disable the RBW processing in the spectrogram.

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:{BANDwidth|BWIDth}[:RESolution]:MODE { OFF | ON | 0 | 1 }
[SENSe]:SGRam[:MEASview<y>]:{BANDwidth|BWIDth}[:RESolution]:MODE?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

OFF or 0 disables the RBW processing. You can select the FFT window using the SENSe:SGRam[:MEASview<y>]:FFT:WINDow (on page 2348) command.

ON or 1 enables the RBW processing. Refer to the SENSe:SGRam:MEASview<y>:BANDwidth:RESolution (on page 2341) command to set the RBW.

Returns

{ OFF | ON | 0 | 1 }

Examples

SENSe:SGRam:MEASview1:BANDwidth:RESolution:MODE ON enables the RBW processing.

[SENSe]:SGRam[:MEASview<y>]:{BANDwidth|BWIDth}:VIDeo

Sets or queries the video bandwidth (VBW). Programming a specified VBW sets SENSe:SGRam[:MEASview<y>]:{BANDwidth|BWIDth}:VIDeo:STATe OFF.

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:{BANDwidth|BWIDth}:VIDeo <value>
[SENSe]:SGRam[:MEASview<y>]:{BANDwidth|BWIDth}:VIDeo?
```

Related Commands

SENSe:SGRam[:MEASview<y>]:{BANDwidth|BWIDth}:VIDeo:STATe (on page 2344)

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

<value>::<NRf> specifies the VBW.

Range: Current RBW/104 (1 Hz minimum) to Current RBW.

Returns

<NRf> specifies the VBW.

Examples

SENSe:SGRam:MEASview1:BANDwidth:VIDeo 200kHz sets the VBW to 200 kHz.

[SENSe]:SGRam[:MEASview<y>]:{BANDwidth|BWIDth}:VIDeo:STATe

Determines whether to enable or disable the video bandwidth (VBW) in the spectrogram measurement.

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:{BANDwidth|BWIDth}:VIDeo:STATe { OFF | ON | 0 | 1 }
[SENSe]:SGRam[:MEASview<y>]:{BANDwidth|BWIDth}:VIDeo:STATe?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

OFF or 0 disables the VBW.

ON or 1 enables the VBW.

Returns

{ OFF | ON | 0 | 1 }

Examples

SENSe:SGRam:MEASview1:BANDwidth:VIDeo:STATe ON enables the VBW.

[SENSe]:SGRam[:MEASview<y>]:{BANDwidth|BWIDth}:OPTimization

Sets or queries the method of optimizing the gain and input bandwidth in the Spectrogram measurement.

Conditions

Measurement view: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:{BANDwidth|BWIDth}:OPTimization { BEST | AUTO | MAXDynrange | MINNoise
   | MINTime }
[SENSe]:SGRam[:MEASview<y>]:{BANDwidth|BWIDth}:OPTimization?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

BEST - BEST (Best for multiple displays) causes the instrument to evaluate all the open displays and select an Acquisition Bandwidth wide enough to satisfy as many measurements as possible

AUTO - AUTO sets the gain and input bandwidth for an optimized trade-off of the competing performance choices.

MAXDynrange - MAXDynrange optimizes the gain and input bandwidth to maximize the dynamic range.

MINNoise - MINNoise optimizes the gain and input bandwidth to minimize noise.

MINTime - MINTime optimizes the gain and input bandwidth to minimize sweep time.

Returns

{ BEST | AUTO | MAXDynrange | MINNoise | MINTime }

Examples

SENSe:SGRam:MEASview1:BANDwidth:OPTimization? might return AUTO, which optimizes automatically the gain and input.

[SENSe]:SGRam[:MEASview<y>]:COLor

Sets or queries the color palette of three-dimensional graphs.

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:COLor { RED | GREen | BLUe | CYAN | BCYan| YELLow | MAGenta | GRAY |
   TEMPerature | SPECtral }
[SENSe]:SGRam[:MEASview<y>]:COLor?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

The following table lists the arguments.

Color palette for spectrogram

Argument	Palette
RED	Red
GREen	Green
BLUe	Blue
CYAN	Cyan
BCYan	Binary cyan
YELLow	Yellow
MAGenta	Magenta
GRAY	Gray
TEMPerature	Temperature
SPECtral	Spectral

Returns

{ RED | GREen | BLUe | CYAN | BCYan| YELLow | MAGenta | GRAY | TEMPerature | SPECtral }

Examples

`SENSe:SGRam:MEASview1:COLor TEMPerature` selects the temperature color palette.

[SENSe]:SGRam[:MEASview<y>]:COLor:MAXimum

Sets or queries the maximum value of the color axis in the spectrogram.

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:COLor:MAXimum <value>
[SENSe]:SGRam[:MEASview<y>]:COLor:MAXimum?
```

Related Commands

SENSe:SGRam[:MEASview<y>]:COLor:MINimum (on page 2347)

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

<value>::<NRf> specifies the maximum value of the color axis.

Range: -100 to +100 dBm.

Returns

<NRf> specifies the maximum value of the color axis.

Examples

SENSe:SGRam:MEASview1:COLor:MAXimum 10 sets the maximum value of the color axis to 10 dBm.

[SENSe]:SGRam[:MEASview<y>]:COLor:MINimum

Sets or queries the minimum value of the color axis in the spectrogram.

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:COLor:MINimum <value>
[SENSe]:SGRam[:MEASview<y>]:COLor:MINimum?
```

Related Commands

SENSe:SGRam[:MEASview<y>]:COLor:MAXimum (on page 2346)

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

<value>::<NRf> specifies the minimum value of the color axis.

Range: -100 to +100 dBm.

Returns

<NRf> specifies the minimum value of the color axis.

Examples

SENSe:SGRam:MEASview1:COLor:MINimum 10 sets the minimum value of the color axis to 10 dBm.

[SENSe]:SGRam[MEASview<y>]:COLor:REVerse

Sets or queries reverse color scheme state.

Conditions

Applicable measurement view must be active.

Group

Sense commands

Syntax

`[SENSe]:SGRam:COLor:REVerse {OFF | ON | 0 | 1}`

`[SENSe]:SGRam:COLor:REVerse?`

Related Commands

[SENSe]:SGRam[:MEASview<y>]:COLor (on page 2345)

Arguments

`OFF` or `0` disables Reverse color scheme.

`ON` or `1` enables Reverse color scheme.

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional. By default y is 1.

Returns

`0` disables Reverse color scheme.

`1` enables Reverse color scheme.

Examples

`SENSe:SGRam:MEASview1:COLor:REVerse OFF` disables reverse color scheme.

[SENSe]:SGRam[:MEASview<y>]:CLEar:RESults (No Query Form)

Restarts multi-trace functions (Average and Max/Min Hold).

Conditions

Measurement view: Spectrogram

Group

Sense commands

Syntax

`[SENSe]:SGRam[:MEASview<y>]:CLEar:RESults`

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Examples

`SENSe:SGRam:MEASview1:CLEar:RESults` clears the results of spectrogram and restarts multi-trace functions.

[SENSe]:SGRam[:MEASview<y>]:FFT:WINDow

Sets or queries the FFT window in the spectrogram.This command is equivalent to `SENSe:SGRam[:MEASview<y>]:FILTer:SHAPe`.

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:FFT:WINDow { KAISer | MIL6db | CISPr | BH4B | UNIForm | FLATtop |
    HANNing }
[SENSe]:SGRam[:MEASview<y>]:FFT:WINDow?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

KAISer selects the Kaiser (RBW) window.

MIL6db selects the -6 dB RBW (MIL) window.

CISPr selects the CISPR window.

BH4B selects the Blackman-Harris 4B type window.

UNIForm selects the uniform window.

FLATtop selects the flat-top window.

HANNing selects the Hanning window.

Returns

{ KAISer | MIL6db | CISPr | BH4B | UNIForm | FLATtop | HANNing }

Examples

SENSe:SGRam:MEASview1:FFT:WINDow HANNing selects the Hanning window.

[SENSe]:SGRam[:MEASview<y>]:FILTer[:SHAPe]

Sets or queries the filter shape in the spectrogram. This command is equivalent to SENSe:SGRam[:MEASview<y>]:FFT:WINDow.

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:FILTer[:SHAPe] { KAISer | MIL6db | CISPr | BH4B | UNIForm | FLATtop |
    HANNing }
[SENSe]:SGRam[:MEASview<y>]:FILTer[:SHAPe]?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

KAISer selects the Kaiser (RBW) window.

MIL6db selects the -6 dB RBW (MIL) window.

CISPr selects the CISPR window.

BH4B selects the Blackman-Harris 4B type window.

UNIForm selects the uniform window.

FLATtop selects the flat-top window.

HANNing selects the Hanning window.

Returns

{ KAISer | MIL6db | CISPr | BH4B | UNIForm | FLATtop | HANNing }

Examples

SENSe:SGRam:MEASview1:FILTer:SHAPe HANNing selects the Hanning window.

[SENSe]:SGRam[:MEASview<y>]:FREQuency:CENTer

Sets or queries the center frequency in the spectrogram.

NOTE: The center, start and stop frequencies are set interlocking each other with the following relationships: (start frequency) = (center frequency) - (span)/2 and (stop frequency) = (center frequency) + (span)/2.

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:FREQuency:CENTer <value>
[SENSe]:SGRam[:MEASview<y>]:FREQuency:CENTer?
```

Related Commands

[SENSe]:SGRam[:MEASview<y>]:FREQuency:STARt (on page 2352)

[SENSe]:SGRam[:MEASview<y>]:FREQuency:STOP (on page 2354)

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

<value>::<NRf> specifies the center frequency.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Returns

<NRf> specifies the center frequency.

Examples

SENSe:SGRam:MEASview1:FREQuency:CENTer 7.5GHz sets the center frequency to 7.5 GHz.

[SENSe]:SGRam[:MEASview<y>]:FREQuency:SPAN

Sets or queries the frequency span in the spectrogram.

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:FREQuency:SPAN <value>
[SENSe]:SGRam[:MEASview<y>]:FREQuency:SPAN?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

<value>::<NRf> specifies the frequency span.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Returns

<NRf> specifies the frequency span.

Examples

`SENSe:SGRam:MEASview1:FREQuency:SPAN 20MHz` sets the span to 20 MHz.

[SENSe]:SGRam[:MEASview<y>]:FREQuency:SPAN:BANDwidth[:RESolution]:RATio

Sets or queries the ratio of span to RBW (Resolution Bandwidth) in the spectrogram. This command is valid when `SENSe:SGRam[:MEASview<y>]:BANDwidth:RESolution:AUTO` is set to On.

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:FREQuency:SPAN:BANDwidth[:RESolution]:RATio <value>
[SENSe]:SGRam[:MEASview<y>]:FREQuency:SPAN:BANDwidth[:RESolution]:RATio?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

<value>::<NRf> specifies the ratio of span to RBW. Range: 20 to 1000.

Programming a specified ratio sets the RBW (= span/ratio), which is rounded down to the nearest valid value.

Returns

<NRf> specifies the ratio of span to RBW.

Examples

SENSe:SGRam:MEASview1:FREQuency:SPAN:BANDwidth:RESolution:RATio 200 sets the ratio to 200, setting the RBW to 200 kHz for the span of 40 MHz.

[SENSe]:SGRam[:MEASview<y>]:FREQuency:SPAN:MAXimum (No Query Form)

Sets the frequency range to the maximum real-time span in the spectrogram.

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:FREQuency:SPAN:MAXimum
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Examples

SENSe:SGRam:MEASview1:FREQuency:SPAN:MAXimum sets the frequency range to the maximum real-time span.

[SENSe]:SGRam[:MEASview<y>]:FREQuency:STARt

Sets or queries the measurement start frequency (left edge of the graph) in the spectrogram.

The center, start and stop frequencies are set interlocking each other. Refer to the SENSe:SGRam[:MEASview<y>]:FREQuency:CENTer (on page 2350) command.

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:FREQuency:STARt <value>
[SENSe]:SGRam[:MEASview<y>]:FREQuency:STARt?
```

Related Commands

[SENSe]:SGRam[:MEASview<y>]:FREQuency:STOP (on page 2354)

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

<value>::<NRf> is the measurement start frequency.

Range: (center frequency) ± (span)/2.

Returns

<NRf> specifies the measurement start frequency.

Examples

SENSe:SGRam:MEASview1:FREQuency:STARt 6.95GHz sets the start frequency to 6.95 GHz.

[SENSe]:SGRam[:MEASview<y>]:FREQuency:STEP

Sets or queries the frequency step size. Programming a specified step size sets [SENSe]:SGRam[:MEASview<y>]:FREQuency:STEP:AUTO OFF.

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:FREQuency:STEP <value>
[SENSe]:SGRam[:MEASview<y>]:FREQuency:STEP?
```

Related Commands

[SENSe]:SGRam[:MEASview<y>]:FREQuency:STEP:AUTO (on page 2354)

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

<value>::<NRf> specifies the frequency step size.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Returns

<NRf> specifies the frequency step size.

Examples

SENSe:SGRam:MEASview1:FREQuency:STEP 1kHz sets the frequency step size to 1 kHz.

[SENSe]:SGRam[:MEASview<y>]:FREQuency:STEP:AUTO

Determines whether to set the frequency step size automatically or manually in the spectrogram.

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:FREQuency:STEP:AUTO { OFF | ON | 0 | 1 }
[SENSe]:SGRam[:MEASview<y>]:FREQuency:STEP:AUTO?
```

Arguments

`<y>` = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

`OFF` or `0` specifies that the frequency step size is set manually using the `SENSe:SGRam[:MEASview<y>]:FREQuency:STEPsize` command.

`ON` or `1` specifies that the frequency step size is set automatically.

Returns

```
{ OFF | ON | 0 | 1 }
```

Examples

`SENSe:SGRam:MEASview1:FREQuency:STEP:AUTO ON` specifies that the frequency step size is set automatically.

[SENSe]:SGRam[:MEASview<y>]:FREQuency:STOP

Sets or queries the measurement stop frequency (right edge of the graph) in the spectrogram.

The center, start and stop frequencies are set interlocking each other. Refer to the [SENSe]:SGRam[:MEASview<y>]:FREQuency:CENTer (on page 2350) command.

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:FREQuency:STOP <value>
[SENSe]:SGRam[:MEASview<y>]:FREQuency:STOP?
```

Related Commands

[SENSe]:SGRam[:MEASview<y>]:FREQuency:STARt (on page 2352)

Arguments

`<y>` = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

`<value>::<NRf>` is the measurement stop frequency.

Range: (center frequency) ± (span)/2.

Returns

<NRf> specifies the measurement stop frequency.

Examples

`SENSe:SGRam:MEASview1:FREQuency:STOP 7.05GHz` sets the stop frequency to 7.05 GHz.

[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:MODe

Sets or queries the vertical mode.

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:MODe {NORMal | MONitor}
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:MODe?
```

Arguments

`<y>` = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

`NORMal` indicates the instrument is set to normal mode.

`MONitor` indicates the instrument is set to Spectrum Monitor mode.

Returns

{NORMal | MONitor}

Examples

`SENSe:SGRam:MEASview1:TIME:SCALe:MODe MONITOR` set the instrument to Spectrum Monitor mode.

[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:OVERlap

Sets or queries the Overlap. Note that changing this value may change the time per division value (DISPlay:SGRam[:MEASview<y>]:TIME[:SCALe]:PER:DIVision).

Conditions

Measurement view: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:OVERlap <value>
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:OVERlap?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

<value> = <NRf> specifies the overlap.

Returns

<NRf> specifies the overlap.

Examples

SENSe:SGRam:MEASview1:TIME:SCALe:OVERlap 1.0E-1 sets the overlap to 0.1.

[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:OVERlap: PERCent

Sets or queries the Overlap in percent. Note that changing this value may change the time per division SENSe:SGRam[:MEASview<y>]:TIME[:SCALe]:PER:DIVision value.

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:OVERlap:PERCent <value>
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:OVERlap:PERCent?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

<value>::<NRf> specifies the overlap in percent.

Returns

<NRf> specifies the overlap in percent.

Examples

SENSe:SGRam:MEASview1:TIME:SCALe:OVERlap:PERCent 1.0E1 sets the overlap to 10%.

[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:PER:DIVision

Sets or queries the vertical Time/division value. Note that changing this value may change the DISPlay:SGRam[:MEASview<y>]:TIME:OVERlap:PERCent value.

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:PER:DIVision <value>
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:PER:DIVision?
```

Related Commands

DISPlay:SGRam[:MEASview<y>]:TIME:OVERlap:PERCent (on page 961)

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

<value>::<NRf> specifies the time per division in seconds.

Returns

<NRf> specifies the overlap in percent.

Examples

SENSe:SGRam:MEASview1:TIME:SCALe:PER:DIVision? might return 1.1160000000E-3.

[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:ZOOM

Sets or queries the zoom scale in the scale tab of spectrogram.

Conditions

Measurement view: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:ZOOM <value>
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:ZOOM?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

<value> = <NRf> specifies the zoom scale value.

Returns

<NRf> specifies the overlap.

Examples

SENSe:SGRam:MEASview1:TIME:SCALe:ZOOM 2 sets the zoom scale to 2.

[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:AUTO

Sets the scale to autoscale in the spectrogram.

Conditions

Measurement view: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:AUTO
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Returns

None

Examples

SENSe:SGRam:MEASview1:TIME:SCALe:AUTO sets the scale to auto.

[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:STARt

Sets or queries the start time in the spectrogram.

Conditions

Measurement view: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:STARt <value>
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:STARt?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

<value> = <NRf> specifies the start time in seconds.

Returns

<NRf> specifies the start time in seconds.

Examples

SENSe:SGRam:MEASview1:TIME:SCALe:STARt? might return 1E-6 s.

[SENSe]:SGRam[:MEASview<y>]:SELEcted:SOURce

Returns the Source number of specified Spectrogram trace.

Conditions

Measurement view: Spectrogram

Source cannot be changed when acquisition or analysis is in progress. Stop the acquisition and change the Source.

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:SELEcted:SOURce {"Source 1"|"Source 2"|"Source 3"|"Source 4"}
[SENSe]:SGRam[:MEASview<y>]:SELEcted:SOURce?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Returns

```
{"Source 1"|"Source 2"|"Source 3"|"Source 4"}
```

Examples

`SENSe:SGRam:MEASview1:SELEcted:SOURce?` might return Source 2 for the trace of Spectrogram measurement view 1.

[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:PER:UPDate:MINutes

Sets or queries the Spectrum Monitor mode Time/Update value for minutes.

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:PER:UPDate:MINutes <value>
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:PER:UPDate:MINutes?
```

Arguments

`<y>` = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

`<value>::<NRf>` sets the time/update value in minutes.

Returns

<NRf> specifies the time/update value in minutes.

Examples

`SENSe:SGRam:MEASview1:TIME:SCALe:PER:UPDate:MINutes 1E1` sets the Time/update minutes value to 10.

[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:PER:UPDate:SEConds

Sets or queries the Spectrum Monitor mode Time/Update value for seconds.

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:PER:UPDate:SEConds <value>
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:PER:UPDate:SEConds?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

<value>::<NRf> sets the time/update value in seconds.

Returns

<NRf> specifies the time/update value in seconds.

Examples

SENSe:SGRam:MEASview1:TIME:SCALe:PER:UPDate:SEConds 1E1 sets the Time/update seconds value to 10.

[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:SPECtrums:PERLine (Query Only)

Queries the Spectrums/Line value.

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:SPECtrums:PERLine?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Returns

<NRf> specifies Spectrum/Line value.

Examples

SENSe:SGRam:MEASview1:TIME:SCALe:SPECtrums:PERLine? might return a 1.0000000000 as a response.

[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:STARt:DIVisions

Sets or queries the Vertical offset or Position in divisions.

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:STARt:DIVisions <value>
[SENSe]:SGRam[:MEASview<y>]:TIME[:SCALe]:STARt:DIVisions?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

<value>::<NRf> sets the vertical offset value in divisions. Range is -130,000 to +130,000.

Returns

<NRf> specifies vertical offset value in divisions.

Examples

SENSe:SGRam:MEASview1:TIME:SCALe:STARt:DIVisions 10 sets the vertical offset to 10 divisions

[SENSe]:SGRam[:MEASview<y>]:WATerfall:DIRection

Sets or queries the direction of the Waterfall (3-D spectrogram) display.

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:WATerfall:DIRection { NW | NE }
[SENSe]:SGRam[:MEASview<y>]:WATerfall:DIRection?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

NW sets the display direction to northwest.

NE sets the display direction to northeast.

Returns

{ NW | NE }

Examples

SENSe:SGRam:MEASview1:WATerfall:DIRection NW sets the waterfall display direction to northwest.

[SENSe]:SGRam[:MEASview<y>]:WATerfall:ENABle

Sets or queries the Waterfall (3-D spectrogram) display.

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:WATerfall:ENABle { OFF | ON | 0 | 1 }
[SENSe]:SGRam[:MEASview<y>]:WATerfall:ENABle?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

OFF or 0 disables the Waterfall display.

ON or 1 enables the Waterfall display.

Returns

{ OFF | ON | 0 | 1 }

Examples

SENSe:SGRam:MEASview1:WATerfall:ENABle ON enables the 3-D Waterfall display.

[SENSe]:SGRam[:MEASview<y>]:WATerfall:Y[:SCALe]:AUTO (No Query Form)

Rescales the height axis automatically to fit the waveform to the screen in the 3-D Waterfall display.

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:WATerfall:Y[:SCALe]:AUTO
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Examples

SENSe:SGRam:MEASview1:WATerfall:Y:SCALe:AUTO rescales the height scale automatically to fit the waveform to the 3-D Waterfall display.

[SENSe]:SGRam[:MEASview<y>]:WATerfall:Y[:SCALe]:OFFSet

Sets or queries the height offset (the value at the top edge of the vertical axis) of the 3-D Waterfall display.

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:WATerfall:Y[:SCALe]:OFFSet <value>
[SENSe]:SGRam[:MEASview<y>]:WATerfall:Y[:SCALe]:OFFSet?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

<value>::<NRf> specifies the height (vertical) offset. Range: -270 to +150 dBm.

Returns

<NRf> specifies the height (vertical) offset.

Examples

SENSe:SGRam:MEASview1:WATerfall:Y:SCALe:OFFSET 12.5 sets the offset to 12.5 dB in the 3-D Waterfall display.

[SENSe]:SGRam[:MEASview<y>]:WATerfall:Y[:SCALe]:RESet (No Query Form)

Resets the height scale of the 3-D Waterfall display to the default values: Height position= 0 dB and Height scale = 100 dB

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:WATerfall:Y[:SCALe]:RESet
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Examples

SENSe:SGRam:MEASview1:WATerfall:Y:SCALe:RESET rescales the height scale automatically to fit the waveform to the 3-D Waterfall display.

[SENSe]:SGRam[:MEASview<y>]:WATerfall:Y[:SCALe]

Sets or queries the height (vertical) range of the Waterfall (3-D spectrogram) display.

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam[:MEASview<y>]:WATerfall:Y[:SCALe] <value>
```

```
[SENSe]:SGRam[:MEASview<y>]:WATerfall:Y[:SCALe]?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

<value>::<NRf> specifies the vertical range. Range: 0.1 to 200 dB. The amplitude unit is set by the [SENSe]:POWer:UNITs command.

You can omit the unit in the argument. When you include the unit in the argument, only dB is available. For the amplitude units Watts, Volts, and Amps, omit the unit in the argument.

Returns

<NRf> specifies the vertical range.

Examples

SENSe:SGRam:MEASview1:WATerfall:Y:SCALe 100dB sets the height range to 100 dB in the 3-D Waterfall display.

[SENSe]:SIGNalpath:PRIority

Sets or queries the signal path priority for the measurements.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:SIGNalpath:PRIority { RF | LF }
[SENSe]:SIGNalpath:PRIority?
```

Arguments

RF sets the signal path to high frequency range.

LF sets the signal path to low frequency range.

Examples

SENSe:SIGNALPATH:PRIORITY LF sets the low frequency signal path when measurements are completely within LF frequency range.

[SENSe]:SIGNalvu:ACQuisition:CONTrol:HORizontal

Sets or queries the other acquisition or Horizontal settings. When the checkbox is checked, the SignalVu controls the Horizontal settings of oscilloscope.

Conditions

Should be connected to MSO/LPD/SX to set or query.

Group

Sense commands

Syntax

```
[SENSe]:SIGNalvu:ACQuisition:CONTrol:HORizontal?
[SENSe]:SIGNalvu:ACQuisition:CONTrol:HORizontal { OFF | ON | 0 | 1 }
```

Arguments

`OFF` or 0 disables Horizontal settings checkbox.

`ON` or 1 enables Horizontal settings checkbox.

Returns

0 disables Horizontal settings checkbox.

1 enables Horizontal settings checkbox.

Examples

`SENSe:SIGNalvu:ACQuisition:CONTrol:HORizontal OFF` disables the Horizontal settings from the SignalVu.

[SENSe]:SIGNalvu:ACQuisition:CONTrol:SAMPle:RATE

Sets or queries the Sample rate, when the checkbox is checked, the SignalVu controls the Sample rate of the oscilloscope.

Conditions

Should be connected to MSO/LPD/SX to set or query.

Group

Sense commands

Syntax

```
[SENSe]:SIGNalvu:ACQuisition:CONTrol:SAMPle:RATE?
[SENSe]:SIGNalvu:ACQuisition:CONTrol:SAMPle:RATE { OFF | ON | 0 | 1 }
```

Arguments

`OFF` or 0 disables Sample rate checkbox.

`ON` or 1 enables Sample rate checkbox.

Returns

0 disables Sample rate checkbox.

1 enables Sample rate checkbox.

Examples

`SENSe:SIGNalvu:ACQuisition:CONTrol:SAMPle:RATE OFF` disables the Sample rate from the SignalVu.

[SENSe]:SIGNalvu:ACQuisition:CONTrol:SOURce<x>:STATe

Sets or queries the Source checkbox state of measurement source for RF, I & Q, Diff I & Q data type.

Conditions

Should be connected to MSO/LPD to Set or Query.

Group

Sense commands

Syntax

```
[SENSe]:SIGNalvu:ACQuisition:CONTrol:SOURce<x>:STATe?
[SENSe]:SIGNalvu:ACQuisition:CONTrol:SOURce<x>:STATe { OFF | ON | 0 | 1 }
```

Arguments

<x>=1 to 8 representing Range of source

OFF or 0 disables Source State.

ON or 1 enables Source State.

Returns

0 disables Source State.

1 enables Source State.

Examples

SENSe:SIGNalvu:ACQuisition:CONTrol:SOURce1:STATe OFF disables Source State.

[SENSe]:SIGNalvu:ACQuisition:CONTrol:TRIGger:POSition

Sets or queries the Trigger position, when the checkbox is checked, the SignalVu controls the Trigger position of oscilloscope.

Conditions

Should be connected to MSO/LPD/SX to set or query.

Group

Sense commands

Syntax

```
[SENSe]:SIGNalvu:ACQuisition:CONTrol:TRIGger:POSition?
[SENSe]:SIGNalvu:ACQuisition:CONTrol:TRIGger:POSition { OFF | ON | 0 | 1 }
```

Arguments

OFF or 0 disables Trigger position checkbox.

ON or 1 enables Trigger position checkbox.

Returns

0 disables Trigger position checkbox.

1 enables Trigger position checkbox.

Examples

SENSe:SIGNalvu:ACQuisition:CONTrol:TRIGger:POSition OFF disables the Trigger position from the SignalVu.

[SENSe]:SIGNalvu:ACQuisition:CONTrol:VERTical

Sets or queries the Vertical settings, when the checkbox is checked, the SignalVu controls the vertical settings of the oscilloscope.

Conditions

Should be connected to MSO/LPD/SX to set or query.

Group

Sense commands

Syntax

```
[SENSe]:SIGNalvu:ACQuisition:CONTrol:VERTical?
[SENSe]:SIGNalvu:ACQuisition:CONTrol:VERTical { OFF | ON | 0 | 1 }
```

Arguments

`OFF` or 0 disables Vertical settings checkbox.

`ON` or 1 enables Vertical settings checkbox.

Returns

0 disables Vertical settings checkbox.

1 enables Vertical settings checkbox.

Examples

`SENSe:SIGNalvu:ACQuisition:CONTrol:VERTical OFF` disables Vertical settings from the SignalVu.

[SENSe]:SIGNalvu:ACQuisition:DIGitizer:SAMPle:RATe

Sets or queries the oscilloscope Sample rate from the Acquisition control panel oscilloscope settings of the SignalVu.

Conditions

Should be connected to MSO/LPD/SX to set or query.

Group

Sense commands

Syntax

```
[SENSe]:SIGNalvu:ACQuisition:DIGitizer:SAMPle:RATE?
[SENSe]:SIGNalvu:ACQuisition:DIGitizer:SAMPle:RATE
```

Arguments

`<value>::<NRf>` specifies the sample rate.

Range: 3.125 S/s to 100 GS/s.

Returns

`<NRf>` specifies the sample rate.

Examples

`SENSe:SIGNalvu:ACQuisition:DIGitizer:SAMPle:RATE 50E+9` sets the oscilloscope sample rate to 50 GS/s.

[SENSe]:SIGNalvu:ACQuisition:DIGitizer:SAMPle:RATE:AUTo

Determines whether to set the oscilloscope Sample rate automatically or manually from the Acquisition control panel oscilloscope settings of the SignalVu.

Conditions

Should be connected to MSO/LPD/SX to set or query.

Group

Sense commands

Syntax

```
[SENSe]:SIGNalvu:ACQuisition:DIGitizer:SAMPle:RATE:AUTo?
[SENSe]:SIGNalvu:ACQuisition:DIGitizer:SAMPle:RATE:AUTo { OFF | ON | 0 | 1 }
```

Arguments

`OFF` or 0 specifies the oscilloscope sample rate is set manually.

`ON` or 1 specifies the oscilloscope sample rate is set automatically.

Examples

`SENSe:SIGNalvu:ACQuisition:DIGitizer:SAMPle:RATE:AUTo ON` sets the oscilloscope Sample rate automatically.

[SENSe]:SIGNalvu:ACQuisition:GLOBal:REFlevel

Sets or queries the Global Reference level.

Conditions

Should be connected to MSO/LPD to Set or Query.

Group

Sense commands

Syntax

```
[SENSe]:SIGNalvu:ACQuisition:GLOBal:REFlevel? (Query)
[SENSe]:SIGNalvu:ACQuisition:GLOBal:REFlevel <value>
```

Arguments

`<value>::<NRf>` specifies the global reference level

Range: {-130 to 30} in dBm

Returns

`<NRf>`

Examples

`SENSe:SIGNalvu:ACQuisition:GLOBal:REFlevel?` might return 5, which indicates the global ref level value is 5.

[SENSe]:SIGNalvu:ACQuisition:INPuttype:REFlevel

Sets or queries the Reference Level input type.

Conditions

Should be connected to MSO/LPD to Set or Query.

Group

Sense commands

Syntax

```
[SENSe]:SIGNalvu:ACQuisition:INPuttype:REFlevel? (Query)
[SENSe]:SIGNalvu:ACQuisition:INPuttype:REFlevel {PERChannel | PERSource | GLOBal}
```

Arguments

`PERChannel` sets the Input Reflevel to Per Channel.

`PERSource` sets the Input Reflevel to Per Source.

`GLOBal` sets the Input Reflevel to Global.

Examples

`SENSe:SIGNalvu:ACQuisition:INPuttype:REFlevel PERChannel` sets input reflevel to per channel.

[SENSe]:SIGNalvu:ACQuisition:IQ:CALibrate (No query form)

Starts the IQ calibration.

Conditions

Should be connected to MSO/LPD/SX.

Group

Sense commands

Syntax

```
SENSe:SIGNalvu:ACQuisition:IQ:CALibrate
```

Examples

`SENSe:SIGNalvu:ACQuisition:IQ:CALibrate` starts the IQ calibration.

[:SENSe]:SIGNalvu:ACQuisition:NULL:IQ:OFFSet:CLEAR

Use this command to clear Null IQ offset values that were calculated using Calibrate IQ.

Related Commands

[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:CALIbrate:IQ (on page 2373)

Conditions

Should be connected to MSO/LPD/SX to query.

Group

Sense commands

Syntax

```
[:SENSe]:SIGNalvu:ACQuisition:NULL:IQ:OFFSet:CLEAR
```

Examples

```
:SENSe:SIGNalvu:ACQuisition:NULL:IQ:OFFSet:CLEAR
```

[:SENSe]:SIGNalvu:ACQuisition:PER:CHANnel<x>:COUPling

Sets or queries the per channel coupling of the measurement. The parameter <x>=1 to 8 represents the channel index.

Conditions

Should be connected to MSO/LPD set or query.

This cmd is query only in case of SX.

Group

Sense commands

Syntax

```
:SENSe:SIGNalvu:ACQuisition:PER:CHANnel<x>:COUPling {AC | DC}
:SENSe:SIGNalvu:ACQuisition:PER:CHANnel<x>:COUPling? (Query)
```

Argument

`AC` sets the specified channel to AC coupling.

`DC` sets the specified channel to DC coupling.

Examples

`SENSe:SIGNalvu:ACQuisition:PER:CHANnel1:COUPling AC` sets coupling of channel 1 to AC.

[:SENSe]:SIGNalvu:ACQuisition:PER:CHANnel<x>:DESkew

Sets or queries the per channel Deskew of the measurement. The parameter <x>=1 to 8 represents the channel index.

Conditions

Should be connected to MSO/LPD/SX to set or query.

Group

Sense commands

Syntax

```
[:SENSe]:SIGNalvu:ACQuisition:PER:CHANnel<x>:DESkew <value>
[:SENSe]:SIGNalvu:ACQuisition:PER:CHANnel<x>:DESkew? (Query)
```

Argument

`<value>::<NRf>` specifies the deskew.

Returns

`<NRf>`

Examples

`SENSe:SIGNalvu:ACQuisition:PER:CHANnel:DESkew:RF 125.0000000000E-9` sets the deskew of channel measurement to 125 ns.

[:SENSe]:SIGNalvu:ACQuisition:PER:CHANnel<x>:EXTattn

Sets or queries the per channel External attenuation of the measurement. The parameter <x>=1 to 8 represents the channel index.

Conditions

Should be connected to MSO/LPD/SX to set or query.

Group

Sense commands

Syntax

```
[:SENSe]:SIGNalvu:ACQuisition:PER:CHANnel<x>:EXTattn <value>
[:SENSe]:SIGNalvu:ACQuisition:PER:CHANnel<x>:EXTattn? (Query)
```

Argument

`<value>::<NRf>` specifies the external attenuation.

Returns

`<NRf>`

Examples

`SENSe:SIGNalvu:ACQuisition:PER:CHANnel1:EXTattn:RF 30.0000000000` sets external attenuation of channel measurement to 30 dB.

SENSe:SIGNalvu:ACQuisition:PER:CHANnel<x>:INSTrument:BWIDth (Query only)

Queries the per channel instrument bandwidth of the measurement. The parameter <x>=1 to 8 represents the channel index.

Conditions

Should be connected to MSO/LPD/SX to set or query.

Group

Sense commands

```
Syntax
SENSe:SIGNalvu:ACQuisition:PER:CHANnel<x>:INSTrument:BWIDth? (Query)
```

Returns

`<NRf>`

Examples

`SENSe:SIGNalvu:ACQuisition:PER:CHANnel1:INSTrument:BWIDth? 500.0000E+06` indicates that instrument bandwidth of channel 1 is 500 MHz.

[:SENSe]:SIGNalvu:ACQuisition:PER:CHANnel<x>:REFlevel

Sets or queries per channel reference level of the measurement. The parameter <x>=1 to 8 represents the channel index.

Conditions

Should be connected to MSO/LPD/SX to set or query.

Group

Sense commands

Syntax

```
[:SENSe]:SIGNalvu:ACQuisition:PER:CHANnel<x>:REFlevel<value>
```

[:SENSe]:SIGNalvu:ACQuisition:PER:CHANnel<x>:REFlevel?(Query)

Argument

<value>::<NRf> specifies the reference level.

Returns

<NRf>

Examples

SENSe:SIGNalvu:ACQuisition:PER:CHANnel1:REFlevel 20.0000000000, sets the reference level of the channel 1 measurement to 20 dB.

[SENSe]:SIGNalvu:ACQuisition:PER:CHANnel<x>:TERMination

Sets or queries the per channel termination of the measurement. The parameter <x>=1 to 8 represents the channel index.

Conditions

Should be connected to MSO/LPD/SX to set or query.

Group

Sense commands

Syntax

SENSe:SIGNalvu:ACQuisition:PER:CHANnel<x>:TERMination {FIFTyohms|ONEMegohms}
SENSe:SIGNalvu:ACQuisition:PER:CHANnel<x>:TERMination? (Query)

Argument

FIFTyohms sets the channel input resistance as 50 Ω.

ONEMegaohms sets the channel input resistance to 1 MΩ.

Returns

<NRf>

Examples

SENSe:SIGNalvu:ACQuisition:PER:CHANnel<x>:TERMination FIFTyohms sets the termination of the Channel 1 to 50 ohms.

[SENSe]:SIGNalvu:ACQuisition:RESet (No query form)

Clears the acquisition data and restarts the SignalVu acquisitions from the oscilloscope.

Conditions

Should be connected to MSO/LPD/SX.

Group

Sense commands

Syntax

SENSe:SIGNalvu:ACQuisition:RESet

Examples

`SENSe:SIGNalvu:ACQuisition:RESet` resets the Signalvu acquisitions from the oscilloscope.

[:SENSe]:SIGNalvu:ACQuisition:SELEcted:CHANnel:TYPE

Sets or queries the selected channel type of the measurement.

Conditions

Should be connected to SX to set or query.

Group

Sense commands

Syntax

```
[:SENSe]:SIGNalvu:ACQuisition:SELEcted:CHANnel:TYPE? (Query)
[:SENSe]:SIGNalvu:ACQuisition:SELEcted:CHANnel:TYPE {ATI | NONAti}
```

Arguments

`ATI` sets the channel type to ATI.

`NONAti` sets the channel type to NON ATI.

Examples

`:SENSe:SIGNalvu:ACQuisition:SELEcted:CHANnel:TYPE ATI`, sets the selected channel type of measurement to ATI.

[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:CALIbrate:IQ

Use this command to perform null IQ offset operation for each source. It is effective from the next acquisition.

The parameter <x>=1 to 8 represents the number of sources.

Related Commands

SENSe:SIGNalvu:ACQuisition:NULL:IQ:OFFSet:CLEAR (on page 2369)

Conditions

Should be connected to MSO/LPD/SX to query.

Group

Sense commands

Syntax

```
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:CALIbrate:IQ
```

Examples

`SENSe:SIGNalvu:ACQuisition:SOURce1:CALIbrate:IQ`

[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:CHANnel[:RF]

Sets or queries the channel of measurement source for RF data type.

Conditions

Should be connected to MSO/LPD to Set or Query.

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:CHANnel[:RF] {"CH1" | "CH2" | "CH3" | "CH4"| "CH5" | "CH6"
    | "CH7" | "CH8"}
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:CHANnel[:RF]? (Query)
```

Arguments

The parameter <x>=1 to 8 representing Range of Source.

Examples

`SENSe:SIGNalvu:ACQuisition:SOURce1:CHANnel:RF?` might return `CH1`, which indicates the channel of measurement source for RF data type.

[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:CHANnel:I

Sets or queries the channel of measurement source for I data type.

Conditions

Should be connected to MSO/LPD to Set or Query.

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:CHANnel:I {"CH1" | "CH2" | "CH3" | "CH4"| "CH5" | "CH6" |
    "CH7"|"CH8"}
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:CHANnel:I? (Query)
```

Arguments

The parameter <x>=1 to 8 representing Range of Source.

Examples

`SENSe:SIGNalvu:ACQuisition:SOURce1:CHANnel:I?` might return `CH1`, which indicates the channel of measurement source for I data type.

[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:CHANnel:Q

Sets or queries the channel of measurement source for Q data type.

Conditions

Should be connected to MSO/LPD to Set or Query.

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:CHANnel:Q {"CH1" | "CH2" | "CH3" | "CH4"| "CH5" | "CH6" |
    "CH7" | "CH8"}
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:CHANnel:Q? (Query)
```

Arguments

The parameter <x>=1 to 8 representing Range of source.

Examples

`SENSe:SIGNalvu:ACQuisition:SOURce1:CHANnel:Q?` might return `CH1`, which indicates the channel of measurement source for Q data type.

[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:CHANnel:IPLus

Sets or queries the channel of measurement source for IPlus data type .

Conditions

Should be connected to MSO/LPD to Set or Query.

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:CHANnel:IPLus {"CH1" | "CH2" | "CH3" | "CH4"| "CH5" |
    "CH6" | "CH7" | "CH8"}
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:CHANnel:IPLus? (Query)
```

Arguments

The parameter <x>=1 to 8 representing Range of source.

Examples

`SENSe:SIGNalvu:ACQuisition:SOURce1:CHANnel:IPLus?` might return `CH1`, which indicates the channel of measurement source IPlus data type.

[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:CHANnel:IMINus

Sets or queries the channel of measurement source for IMinus data type.

Conditions

Should be connected to MSO/LPD to Set or Query.

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:CHANnel:IMINus {"CH1" | "CH2" | "CH3" | "CH4"| "CH5" |
    "CH6" | "CH7" | "CH8"}
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:CHANnel:IMINus? (Query)
```

Arguments

The parameter <x>=1 to 8 representing Range of source.

Examples

`SENSe:SIGNalvu:ACQuisition:SOURce1:CHANnel:IMINus?` might return `CH1`, which indicates the channel of measurement source for IMinus data type.

[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:CHANnel:QPLus

Sets or queries the channel of measurement source for QPlus data type.

Conditions

Should be connected to MSO/LPD to Set or Query.

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:CHANnel:QPLus {"CH1" | "CH2" | "CH3" | "CH4"| "CH5" |
   "CH6" | "CH7" | "CH8" }
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:CHANnel:QPLus? (Query)
```

Arguments

The parameter `<x>`=1 to 8 representing Range of source.

Examples

`SENSe:SIGNalvu:ACQuisition:SOURce1:CHANnel:QPLus?` might return `CH1`, which indicates the channel of measurement source for QPlus data type.

[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:CHANnel:QMINus

Sets or queries the channel of measurement source for QMinus data type.

Conditions

Should be connected to MSO/LPD to Set or Query.

Source keyword is optional, if not present Source1 will be default source.

Group

Sense commands

Syntax

```
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:CHANnel:QMINus {"CH1" | "CH2" | "CH3" | "CH4"| "CH5" |
   "CH6" | "CH7"|"CH8"
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:CHANnel:QMINus? (Query)
```

Arguments

The parameter `<x>`=1 to 8 representing Range of source.

Examples

`SENSe:SIGNalvu:ACQuisition:SOURce1:CHANnel:QMINus?` might return `CH1`, which indicates the channel of measurement source for QMinus data type.

[:SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]: DOWNconverter:INJEctionside

Sets or queries the downconverter injection side. The parameter <x>=1 to 8 represents the range of the source.

Conditions

Should be connected to MSO/LPD/RSA/SX to set or query.

Source keyword is optional, if not present, Source1 will be default source.

Group

Sense commands

Syntax

```
:SENSe:SIGNalvu:ACQuisition:SOURce<x>:DOWNconverter:INJEctionside {LOWI | HIGH }
:SENSe:SIGNalvu:ACQuisition:SOURce<x>:DOWNconverter:INJEctionside? (Query)
```

Examples

`ACQuisition:SOURce1:DOWNconverter:INJEctionside?` might return `LOWI`, which indicates the downconverter injection side set to low.

[:SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]: DOWNconverter:INSTrument:BANDwidth:MAXFreq

Sets or queries the maximum range of the instrument bandwidth. The parameter <x>=1 to 8 represents the range of the source.

Conditions

Should be connected to MSO/LPD/RSA/SX to set or query.

Source keyword is optional, if not present, Source1 will be default source.

Group

Sense commands

Syntax

```
[:SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:DOWNconverter:INSTrument:BANDwidth:MAXFreq <value>
[:SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:DOWNconverter:INSTrument:BANDwidth:MAXFreq? (Query)
```

Argument

`<value>::<NRf>` specifies the minimum value of the instrument bandwidth being used.

Returns

`<NRf>`

Examples

`SENSe:SIGNalvu:ACQuisition:SOURce1:DOWNconverter:INSTrument:BANDwidth:MaxFreq?` might return `7.5000000000E+9`, which indicates the maximum range of the instrument bandwidth.

[:SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]: DOWNconverter:INSTrument:BANDwidth:MINFreq

Sets or queries the minimum range of instrument bandwidth. The parameter <x>=1 to 8 represents the range of the source.

Conditions

Should be connected to MSO/LPD/RSA/SX to set or query.

Source keyword is optional, if not present, Source1 will be default source.

Group

Sense commands

Syntax

```
[:SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:DOWNconverter:INSTrument:BANDwidth:MINFreq <value>
[:SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:DOWNconverter:INSTrument:BANDwidth:MINFreq? (Query)
```

Argument

`<value>::<NRf>` specifies the minimum value of the instrument bandwidth being used.

Returns

`<NRf>`

Examples

`SENSe:SIGNalvu:ACQuisition:SOURce1:DOWNconverter:INSTrument:BANDwidth:MINFreq?` might return `0 Hz`, which indicates the minimum range of the instrument bandwidth.

[:SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]: DOWNconverter:LOFReq

Sets or queries the downconverter LO frequency. The parameter <x>=1 to 8 represents the range of the source.

Conditions

Should be connected to MSO/LPD/RSA/SX to set or query.

Source keyword is optional, if not present, Source1 will be default source.

Syntax

```
[:SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:DOWNconverter:LOFReq <value>
[:Senses]:SIGNalvu:ACQuisition[:SOURce<x>]:DOWNconverter:LOFReq? (Query)
```

Argument

`<value>::<NRf>` specifies the LO frequency.

Returns

<NRf>

Examples

`:SENSe:SIGNalvu:ACQuisition:SOURce1:DOWNconverter:LOFReq 3.0000000000E+9` sets the LO frequency to 3 GHz.

[:SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:DOWNconverter:STATe

Sets or queries the downconverter state. The parameter <x>=1 to 8 represents the range of the source.

Conditions

Should be connected to MSO/LPD/RSA/SX to set or query.

Source keyword is optional, if not present, Source1 will be default source.

Group

Sense commands

Syntax

```
[:SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:DOWNconverter:STATe{ TRUE | FALSE}
[:SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:DOWNconverter:STATe? (Query)
```

Examples

`:SENSe:SIGNalvu:ACQuisition:SOURce1:DOWNconverter:STATe TRUE` enables the downconverter for Source 1.

[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:PERSrc:REFlevel

Sets or queries the Reference level Per source .

Conditions

Should be connected to MSO/LPD to Set or Query.

Group

Sense comands

Syntax

```
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:PERSrc:REFlevel? (Query)
[SENSe]:SIGNalvu:ACQuisition[:SOURce<x>]:PERSrc:REFlevel <Value>
```

Arguments

<value>::<NRf> specifies the global reference level

Range: {-130 to 30} in dBm

Returns

`<NRf>`

Examples

`SENSe:SIGNalvu:ACQuisition:SOURce1:PERSrc:REFlevel?` might return `5`, which indicates the ref level value of source 1 is 5.

[SENSe]:SPECtrum[:MEASview<y>]:SELEcted:SOURce

Sets or queries the Source of the selected Measurement view in pulse trace

Conditions

Measurement views: Spectrum with specified Measurement view number

Source cannot be changed when acquisition or analysis is in progress. Stop the acquisition and change the Source.

Group

Sense commands

Syntax

```
[SENSe]:SPECtrum[:MEASview<y>]:SELEcted:SOURce {"Source1" | "Source2" | ""Source3" | "Source4" }
[SENSe]:SPECtrum[:MEASview<y>]:SELEcted:SOURce?
```

Arguments

<y>=1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1

Examples

:SENSe:SPECtrum:TRACE:MEASVIEW1:SELECTED:SOURCE? queries the Source of the Spectrum Measurement View 1.

[SENSe]:SPECtrum[:MEASview<y>]:{BANDwidth|BWIDth}[:RESolution]

Sets or queries the resolution bandwidth (RBW). Programming a specified RBW sets [SENSe]:SPECtrum[:MEASview<y>]:{BANDwidth|BWIDth}[:RESolution]:AUTO OFF.

Conditions

Measurement views: Spectrum

Group

Sense commands

Syntax

```
[SENSe]:SPECtrum[:MEASview<y>]:{BANDwidth|BWIDth}[:RESolution] <value>
[SENSe]:SPECtrum[:MEASview<y>[:{BANDwidth|BWIDth}[:RESolution]?
```

Related Commands

[SENSe]:SPECtrum[:MEASview<y>]:{BANDwidth|BWIDth}[:RESolution]:AUTO (on page 2381)

Arguments

<value>::<NRf> specifies the RBW. Range: 1 Hz to 10 MHz.

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

SENSE:SPECTRUM:MEASVIEW2:BANDWIDTH:RESOLUTION 200kHz sets the RBW to 200 kHz.

[SENSe]:SPECtrum[:MEASview<y>]:{BANDwidth|BWIDth}[:RESolution]:ACTual? (Query Only)

Queries the actual resolution bandwidth (RBW) in the spectrum measurement.

Conditions

Measurement views: Spectrum

Group

Sense commands

Syntax

```
[SENSe]:SPECtrum[:MEASview<y>]:{BANDwidth|BWIDth}[:RESolution]:ACTual?
```

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

`<NRf>` The actual RBW in Hz.

Examples

`SENSE:SPECTRUM:MEASVIEW2:BANDWIDTH:RESOLUTION:ACTUAL?` might return `299.624E+3`, indicating that the actual RBW is 299.624 kHz.

[SENSe]:SPECtrum[:MEASview<y>]:{BANDwidth|BWIDth}[:RESolution]:AUTO

Determines whether to set the resolution bandwidth (RBW) automatically or manually in the spectrum measurement.

Conditions

Measurement views: Spectrum

Group

Sense commands

Syntax

```
[SENSe]:SPECtrum[:MEASview<y>]:{BANDwidth|BWIDth}[:RESolution]:AUTO { OFF| ON | 0 | 1 }
[SENSe]:SPECtrum[:MEASview<y>]:{BANDwidth|BWIDth}[:RESolution]:AUTO?
```

Arguments

`OFF` or `0` specifies that the resolution bandwidth is set manually using the `[SENSe]:SPECtrum[:MEASview<y>]:{BANDwidth|BWIDth}[:RESolution]` command.

`ON` or `1` specifies that the resolution bandwidth is set automatically.

The parameter `<y>`=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`SENSE:SPECTRUM:MEASVIEW2:BANDWIDTH:RESOLUTION:AUTO ON` sets the resolution bandwidth automatically.

[SENSe]:SPECtrum[:MEASview<y>]:{BANDwidth|BWIDth}[:RESolution]:MODE

Determines whether to enable or disable the RBW process.

NOTE: This is an obsolete command.

Conditions

Measurement views: Spectrum

Group

Sense commands

Syntax

```
[SENSe]:SPECtrum[:MEASview<y>]:{BANDwidth|BWIDth}[:RESolution]:MODE { OFF| ON | 0 | 1 }
[SENSe]:SPECtrum[:MEASview<y>]:{BANDwidth|BWIDth}[:RESolution]:MODE?
```

Arguments

OFF or 0 disables the RBW process. You can select the FFT window using the `[SENSe]:SPECtrum[:MEASview<y>]:FFT:WINDow` command.

ON or 1 enables the RBW process. Refer to the `[SENSe]:SPECtrum[:MEASview<y>]:{BANDwidth|BWIDth}[:RESolution]` command to set the RBW.

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`SENSE:SPECTRUM:MEASVIEW2:BANDWIDTH:RESOLUTION:MODE ON` enables the RBW process.

[SENSe]:SPECtrum[:MEASview<y>]:{BANDwidth|BWIDth}:VIDeo

Sets or queries the video bandwidth (VBW). Programming a specified VBW sets `[SENSe]:SPECtrum[:MEASview<y>]:{BANDwidth|BWIDth}:VIDeo:STATe OFF`.

Conditions

Measurement views: Spectrum

Group

Sense commands

Syntax

```
[SENSe]:SPECtrum[:MEASview<y>]:{BANDwidth|BWIDth}:VIDeo <value>
[SENSe]:SPECtrum[:MEASview<y>]:{BANDwidth|BWIDth}:VIDeo?
```

Arguments

`<value>::<NRf>` specifies the VBW.

Range: Current RBW/104 (1 Hz minimum) to Current RBW.

The parameter `<y>`=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`SENSE:SPECTRUM:MEASVIEW2:BANDWIDTH:VIDEO 200kHz` sets the VBW to 200 kHz.

[SENSe]:SPECtrum[:MEASview<y>]:{BANDwidth|BWIDth}:VIDeo:STATe

Determines whether to enable or disable the video bandwidth (VBW) in the spectrum measurement.

Conditions

Measurement views: Spectrum

Group

Sense commands

Syntax

```
[SENSe]:SPECtrum[:MEASview<y>]:{BANDwidth|BWIDth}:VIDeo:STATe { OFF | ON| 0 | 1 }
[SENSe]:SPECtrum[:MEASview<y>]:{BANDwidth|BWIDth}:VIDeo:STATe?
```

Arguments

`OFF` or `0` disables the VBW.

`ON` or `1` enables the VBW.

The parameter `<y>`=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`SENSE:SPECTRUM:MEASVIEW2:BANDWIDTH:VIDEO:STATE ON` enables the VBW.

[SENSe]:SPECtrum[:MEASview<y>]:CLEar:RESults (No Query Form)

Restarts multi-trace functions (Average and Max/Min Hold).

Conditions

Measurement views: Spectrum

Group

Sense commands

Syntax

```
[SENSe]:SPECtrum[:MEASview<y>]:CLEar:RESults
```

Arguments

The parameter `<y>`=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`SENSe:SPECtrum:MEASview2:CLEar:RESults` restarts multi-trace functions.

[SENSe]:SPECtrum[:MEASview<y>]:COLor

Sets or queries the color palette of three-dimensional graphs.

Conditions

Measurement views: Spectrum

Group

Sense commands

Syntax

```
[SENSe]:SPECtrum[:MEASview<y>]:COLor {RED | GREen | BLUe | CYAN | BCYan | YELLow |
MAGenta | GRAY | TEMPerature | SPECtral}
```

```
[SENSe]:SPECtrum[:MEASview<y>]:COLor?
```

Arguments

The following table lists the arguments.

Argument	Description
`RED`	Red
`GREen`	Green
`BLUe`	Blue
`CYAN`	Cyan
`BCYan`	Binary cyan
`YELLow`	Yellow
`MAGenta`	Magenta
`GRAY`	Gray
`TEMPerature`	Temperature
`SPECtral`	Spectral

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`SENSE:SPECTRUM:COLOR TEMPERATURE` selects the temperature color palette.

[SENSe]:SPECtrum[:MEASview<y>]:COLor:MAXimum

Sets or queries the maximum value of the color axis in the Spectrum measurement.

Conditions

Measurement views: Spectrum

Group

Sense commands

Syntax

`[SENSe]:SPECtrum[:MEASview<y>]:COLor:MAXimum <value>`

`[SENSe]:SPECtrum[:MEASview<y>]:COLor:MAXimum?`

Arguments

`<value>::=<NRf>` specifies the maximum value of the color axis.

Range: The minimum value to 100%.

The minimum value is set using the `[SENSe]:SPECtrum[:MEASview<y>]:COLor:MINimum` command.

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`SENSE:SPECTRUM:COLOR:MAXIMUM 90` sets the maximum value of the color axis to 90%.

[SENSe]:SPECtrum[:MEASview<y>]:COLor:MINimum

Sets or queries the minimum value of the color axis in the Spectrum measurement.

Conditions

Measurement views: Spectrum

Group

Sense commands

Syntax

`[SENSe]:SPECtrum[:MEASview<y>]:COLor:MINimum <value>`

`[SENSe]:SPECtrum[:MEASview<y>]:COLor:MINimum?`

Arguments

`<value>::=<NRf>` specifies the minimum value of the color axis.

Range: 0% to the maximum value.

The maximum value is set using the `[SENSe]:SPECtrum[:MEASview<y>]:COLor:MAXimum` command.

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`SENSE:SPECTRUM:COLOR:MINIMUM 10` sets the minimum value of the color axis to 10%.

[SENSe]:SPECtrum[:MEASview<y>]:COLor:REVerse[:STATe]

Determines whether to reverse the color palette in the Spectrum view. When reversed, the highest density takes the color normally used for the lowest.

Conditions

Measurement views: Spectrum

Group

Sense commands

Syntax

```
[SENSe]:SPECtrum[:MEASview<y>]:COLor:REVerse[:STATe] {OFF | ON | 0 | 1}
[SENSe]:SPECtrum[:MEASview<y>]:COLor:REVerse[:STATe]?
```

Arguments

`OFF` or `0` uses the color palette in its normal order.

`ON` or `1` reverses the color palette.

The parameter <y>= 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`SENSE:SPECTRUM:COLOR:REVERSE:STATE ON` reverses the color palette.

[SENSe]:SPECtrum[:MEASview<y>]:FFT:WINDow

Sets or queries the FFT window in the spectrum measurement.This command is equivalent to `[SENSe]:SPECtrum[:MEASview<y>]:FILTer[:SHAPe]`.

Conditions

Measurement views: Spectrum

Group

Sense commands

Syntax

```
[SENSe]:SPECtrum[:MEASview<y>]:FFT:WINDow { KAISer | MIL6db | CISPr | BH4B| UNIForm | FLATtop |
   HANNing }
[SENSe]:SPECtrum[:MEASview<y>]:FFT:WINDow?
```

Arguments

`KAISer` selects the Kaiser (RBW) window.

`MIL6db` selects the -6 dB RBW (MIL) window.

`CISPr` selects the CISPR window.

`BH4B` selects the Blackman-Harris 4B type window.

`UNIForm` selects the uniform window.

`FLATtop` selects the flat-top window.

`HANNing` selects the Hanning window.

The parameter `<y>`=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`SENSE:SPECTRUM:MEASVIEW2:FFT:WINDOW HANNing` selects the Hanning window.

[SENSe]:SPECtrum[:MEASview<y>]:FILTer[:SHAPe]

Sets or queries the filter shape in the spectrum measurement. This command is equivalent to `[SENSe]:SPECtrum[:MEASview<y>]:FFT:WINDow`.

Conditions

Measurement views: Spectrum

Group

Sense commands

Syntax

```
[SENSe]:SPECtrum[:MEASview<y>]:FILTer[:SHAPe] { KAISer | MIL6db | CISPr | BH4B | UNIForm | FLATtop
    | HANNing }
[SENSe]:SPECtrum[:MEASview<y>]:FILTer[:SHAPe]?
```

Arguments

`KAISer` selects the Kaiser (RBW) window.

`MIL6db` selects the -6 dB RBW (MIL) window.

`CISPr` selects the CISPR window.

`BH4B` selects the Blackman-Harris 4B type window.

`UNIForm` selects the uniform window.

`FLATtop` selects the flat-top window.

`HANNing` selects the Hanning window.

Examples

`SENSe:SPECTRUM:MEASVIEW1:FILTER:SHAPE HANNing` selects the Hanning window.

[SENSe]:SPECtrum[:MEASview<y>]:FREQuency:CENTer

Sets or queries the center frequency in the spectrum measurement.

NOTE: The center, start and stop frequencies are set interlocking each other with the following relationships: (start frequency) = (center frequency) - (span)/2 and (stop frequency) = (center frequency) + (span)/2.

Conditions

Measurement views: Spectrum

Group

Sense commands

Syntax

```
[SENSe]:SPECtrum[:MEASview<y>]:FREQuency:CENTer <value>
[SENSe]:SPECtrum[:MEASview<y>]:FREQuency:CENTer?
```

Related Commands

[SENSe]:SPECtrum[:MEASview<y>]:FREQuency:STARt (on page 2389)

[SENSe]:SPECtrum[:MEASview<y>]:FREQuency:STOP (on page 2391)

Arguments

`<value>::<NRf>` specifies the center frequency.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`SENSE:SPECTRUM:MEASVIEW2:FREQUENCY:CENTER 7.5GHz` sets the center frequency to 7.5 GHz..

[SENSe]:SPECtrum[:MEASview<y>]:FREQuency:SPAN

Sets or queries the frequency span in the spectrum measurement.

Conditions

Measurement views: Spectrum

Group

Sense commands

Syntax

```
[SENSe]:SPECtrum[:MEASview<y>]:FREQuency:SPAN <value>
[SENSe]:SPECtrum[:MEASview<y>]:FREQuency:SPAN?
```

Arguments

`<value>::<NRf>` specifies the frequency span.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

The parameter `<y>`=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`SENSe:SPECtrum:MEASview2:FREQuency:SPAN 20MHz` sets the span to 20 MHz

[SENSe]:SPECtrum[:MEASview<y>]:FREQuency:SPAN:BANDwidth[:RESolution]:RATio

Sets or queries the ratio of span to RBW (Resolution Bandwidth) in the spectrum measurement. This command is valid when `[SENSe]:SPECtrum[:MEASview<y>]:{BANDwidth|BWIDth}[:RESolution]:AUTO` is set to On.

Conditions

Measurement views: Spectrum

Group

Sense commands

Syntax

```
[SENSe]:SPECtrum[:MEASview<y>]:FREQuency:SPAN:BANDwidth[:RESolution]:RATio <value>
[SENSe]:SPECtrum[:MEASview<y>]:FREQuency:SPAN:BANDwidth[:RESolution]:RATio?
```

Arguments

`<value>::<NRf>` specifies the ratio of span to RBW. Range: 20 to 1000. Programming a specified ratio sets the RBW (= span/ratio), which is rounded down to the nearest valid value.

The parameter `<y>`=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`SENSe:SPECtrum:MEASview2:FREQuency:SPAN:BANDwidth:RESolution:RATio 200` sets the ratio to 200, setting the RBW to 200 kHz for the span of 40 MHz.

[SENSe]:SPECtrum[:MEASview<y>]:FREQuency:STARt

Sets or queries the measurement start frequency (left edge on the graph) in the spectrum measurement.

The center, start and stop frequencies are set interlocking each other. Refer to the [SENSe]:SPECtrum[:MEASview<y>]:FREQuency:CENTer (on page 2387) command.

Conditions

Measurement views: Spectrum

Group

Sense commands

Syntax

```
[SENSe]:SPECtrum[:MEASview<y>]:FREQuency:STARt <value>
[SENSe]:SPECtrum[:MEASview<y>]:FREQuency:STARt?
```

Related Commands

[SENSe]:SPECtrum[:MEASview<y>]:FREQuency:STOP (on page 2391)

Arguments

`<value>::<NRf>` is the measurement start frequency.

Range: (center frequency) ±(span)/2.

The parameter `<y>`=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`SENSe:SPECtrum:MEASview2:FREQuency:STARt 6.95GHz` sets the start frequency to 6.95 GHz.

[SENSe]:SPECtrum[:MEASview<y>]:FREQuency:STEP

Sets or queries the frequency step size (the amount per press by which the up or down key changes the setting value). Programming a specified step size sets `[SENSe]:SPECtrum[:MEASview<y>]:FREQuency:STEP:AUTO OFF`.

The center, start and stop frequencies are set interlocking each other. Refer to the [SENSe]:SPECtrum[:MEASview<y>]:FREQuency:CENTer (on page 2387) command.

Conditions

Measurement views: Spectrum

Group

Sense commands

Syntax

```
[SENSe]:SPECtrum[:MEASview<y>]:FREQuency:STEP <value>
[SENSe]:SPECtrum[:MEASview<y>]:FREQuency:STEP?
```

Related Commands

[SENSe]:SPECtrum[:MEASview<y>]:FREQuency:STEP:AUTO (on page 2390)

Arguments

`<value>::<NRf>` specifies the frequency step size.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`SENSe:SPECtrum:MEASview2:FREQuency:STEP 1.5kHz` sets the step size to 1.5 kHz.

[SENSe]:SPECtrum[:MEASview<y>]:FREQuency:STEP:AUTO

Determines whether to set the frequency step size automatically or manually.

Conditions

Measurement views: Spectrum

Group

Sense commands

Syntax

```
[SENSe]:SPECtrum[:MEASview<y>]:FREQuency:STEP:AUTO { OFF | ON | 0 | 1 }
[SENSe]:SPECtrum[:MEASview<y>]:FREQuency:STEP:AUTO?
```

Arguments

`OFF` or `0` specifies that the frequency step size is set manually using the `[SENSe]:SPECtrum[:MEASview<y>]:FREQuency:STEP` command.

ON or 1 specifies that the frequency step size is set automatically.

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

SENSe:SPECtrum:MEASview2:FREQuency:STEP:AUTO ON sets the frequency step size automatically

[SENSe]:SPECtrum[:MEASview<y>]:FREQuency:STOP

Sets or queries the measurement stop frequency (right edge on the graph) in the spectrum measurement.

The center, start and stop frequencies are set interlocking each other. Refer to the [SENSe]:SPECtrum[:MEASview<y>]:FREQuency:CENTer (on page 2387) command.

Conditions

Measurement views: Spectrum

Group

Sense commands

Syntax

```
[SENSe]:SPECtrum[:MEASview<y>]:FREQuency:STOP <value>
[SENSe]:SPECtrum[:MEASview<y>]:FREQuency:STOP?
```

Related Commands

[SENSe]:SPECtrum[:MEASview<y>]:FREQuency:STARt (on page 2389)

Arguments

<value>::<NRf> is the measurement start frequency.

Range: (center frequency) ± (span)/2.

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

SENSe:SPECtrum:MEASview2:FREQuency:STOP 7.05 sets the stop frequency to 7.05 GHz.

[SENSe]:SPECtrum[:MEASview<y>]:LENGth

Sets or queries the spectrum length when [SENSe]:SPECtrum:TIMe:MODe is INDependent. Programming a specified length sets [SENSe]:SPECtrum[:MEASview<y>]:LENGth:AUTO OFF.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:SPECtrum[:MEASview<y>]:LENGth <value>
[SENSe]:SPECtrum[:MEASview<y>]:LENGth?
```

Related Commands

[SENSe]:SPECtrum[:MEASview<y>]:LENGth:AUTO (on page 2392)

[SENSe]:SPECtrum[:MEASview<y>]:STARt (on page 2394)

Arguments

<value>::<NRf> specifies the spectrum length.

Range: 0 to [(acquisition length) - (spectrum start)].

If [(spectrum start) + (spectrum length)] > (acquisition length), the actual spectrum length is reduced to the acquisition length.

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

SENSe:SPECtrum:MEASview2:LENGth 38.5us sets the spectrum length to 38.5 µs.

[SENSe]:SPECtrum[:MEASview<y>]:LENGth:ACTual (Query Only)

Queries the actual spectrum length.

Conditions

Measurement views: All

Group

Sense commands

Syntax

[SENSe]:SPECtrum[:MEASview<y>]:LENGth:ACTual?

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

<NRf> Actual spectrum length in seconds.

Examples

SENSe:SPECtrum:MEASview2:LENGth:ACTual? might return 6.337E-6, indicating that the actual spectrum length is 6.337 µs

[SENSe]:SPECtrum[:MEASview<y>]:LENGth:AUTO

Sets or queries the spectrum length mode (automatic or manual) when [SENSe]:SPECtrum:TIMe:MODe is INDependent.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:SPECtrum[:MEASview<y>]:LENGth:AUTO { OFF | ON | 0 | 1 }
[SENSe]:SPECtrum[:MEASview<y>]:LENGth:AUTO?
```

Arguments

`OFF` or `0` sets the spectrum length manually, using the `[SENSe]:SPECtrum:LENGth` command.

`ON` or `1` sets the spectrum length automatically.

Examples

`SENSe:SPECTRUM:MEASview1:LENGTH:AUTO ON` sets the spectrum length automatically.

[SENSe]:SPECtrum[:MEASview<y>]:MAX:SPAN (No Query Form)

Sets the frequency span to the maximum full span instead of the maximum real-time span. The maximum span depends on the model.

Conditions

Measurement views: Spectrum

Group

Sense commands

Syntax

```
[SENSe]:SPECtrum[:MEASview<y>]:MAX:SPAN
```

Arguments

The parameter `<y>`=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`SENSe:SPECtrum:MEASview2:MAX:SPAN` sets the frequency span to the maximum span.

[SENSe]:SPECtrum[:MEASview<y>]:POINts:COUNt

Sets or queries the number of sample points on the signal spectrum.

Conditions

Measurement views: Spectrum

Group

Sense commands

Syntax

```
[SENSe]:SPECtrum[:MEASview<y>]:POINts:COUNt { P801 | P1601 | P2401 | P3201| P4001 | P6401 | P8001 |
    P10401 }
[SENSe]:SPECtrum[:MEASview<y>]:POINts:COUNt?
```

Arguments

`P801` sets the number of sample points to 801.

P1601 sets the number of sample points to 1601.

P2401 sets the number of sample points to 2401.

P3201 sets the number of sample points to 3201.

P4001 sets the number of sample points to 4001.

P6401 sets the number of sample points to 6401.

P8001 sets the number of sample points to 8001.

P10401 sets the number of sample points to 10401.

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

SENSe:SPECtrum:MEASview2:POINts:COUNt P801 sets the number of sample points to 801.

[SENSe]:SPECtrum[:MEASview<y>]:STARt

Sets or queries the spectrum offset time when [SENSe]:SPECtrum[:MEASview<y>]:TIMe:MODe is INDependent.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:SPECtrum[:MEASview<y>]:STARt <value>
[SENSe]:SPECtrum[:MEASview<y>]:STARt?
```

Related Commands

[SENSe]:SPECtrum[:MEASview<y>]:LENGth (on page 2391)

Arguments

<value>::<NRf> specifies the spectrum offset time.

Range: 0 to the acquisition length.

If [(spectrum start) + (spectrum length)] > (acquisition length), the actual spectrum length is reduced to the acquisition length.

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

SENSe:SPECTRUM:MEASview1:START 23.5us sets the analysis offset to 23.5 µs.

[SENSe]:SPECtrum[:MEASview<y>]:STARt:AUTO

Sets or queries the spectrum offset mode (automatic or manual) when [SENSe]:SPECtrum[:MEASview<y>]:TIMe:MODe is INDependent.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:SPECtrum[:MEASview<y>]:STARt:AUTO { OFF | ON | 0 | 1 }
[SENSe]:SPECtrum[:MEASview<y>]:STARt:AUTO?
```

Arguments

`OFF` or `0` sets the spectrum offset manually.

`ON` or `1` sets the spectrum length automatically.

The parameter `<y>`=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`SENSe:SPECTRUM:MEASview1:START:AUTO ON` sets the spectrum offset automatically.

[SENSe]:SPECtrum[:MEASview<y>]:TIMe:MODe

Sets or queries the spectrum time parameter mode (automatic or manual).

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:SPECtrum[:MEASview<y>]:TIMe:MODe { INDependent | COMMon }
[SENSe]:SPECtrum[:MEASview<y>]:TIMe:MODe?
```

Related Commands

[SENSe]:ANALysis (on page 1955) commands

Arguments

`INDependent` sets the spectrum time parameters manually, using the `[SENSe]:SPECtrum:STARt` and `[SENSe]:SPECtrum:LENGth` commands.

`COMMon` sets the spectrum time parameters automatically using the analysis time settings.

The parameter `<y>`=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Examples

`SENSe:SPECTRUM:MEASview1:TIME:MODE COMMon` sets the spectrum time parameters automatically.

[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth}

Sets or queries the channel bandwidth for the carrier as power reference. This command is valid when `[SENSe]:SPURious:REFerence` is set to `CARRier`.

Conditions

Measurement views: Spurious

Group

Sense commands

Syntax

```
[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth} <value>
[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth}?
```

Arguments

`<value>::<NRf>` specifies the channel bandwidth.

Range: 100 Hz to 40 MHz (Standard) / 110 MHz (Option 110).

Examples

`SENSe:SPURIOUS:CARRIER:BANDWIDTH 20MHz` sets the channel bandwidth to 20 MHz.

[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth}:INTegration

Sets or queries the integration bandwidth to calculate the carrier power. This command is valid when `[SENSe]:SPURious:REFerence` is set to `CARRier`.

Conditions

Measurement views: Spurious

Group

Sense commands

Syntax

```
[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth}:INTegration <value>
[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth}:INTegration?
```

Arguments

`<value>::<NRf>` specifies the integration bandwidth.

Range: 100 Hz to 40 MHz (Standard) / 110 MHz (Option 110).

Examples

`SENSe:SPURIOUS:CARRIER:BANDWIDTH:INTEGRATION 2MHz` sets the integration bandwidth to 2 MHz.

[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth}[:RESolution]

Sets or queries the resolution bandwidth (RBW) to measure the carrier power. This command is valid when `[SENSe]:SPURious:REFerence` is set to `CARRier`. Programming a specified RBW sets `[SENSe]:SPURious:CARRier:BANDwidth[RESolution]:AUTO` OFF.

Conditions

Measurement views: Spurious

Group

Sense commands

Syntax

```
[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth}[:RESolution] <value>
[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth}[:RESolution]?
```

Related Commands

[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth}[:RESolution]:AUTO (on page 2397)

Arguments

`<value>::<NRf>` specifies the RBW. Range: 1 Hz to 10 MHz.

Examples

`SENSe:SPURIOUS:CARRIER:BANDWIDTH:RESOLUTION 200kHz` sets the RBW to 200 kHz.

[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth}[:RESolution]:AUTO

Determines whether to set the resolution bandwidth (RBW) automatically or manually when the power reference is set to carrier (`[SENSe]:SPURious:REFerence` is set to `CARRier`) in the Spurious measurement.

Conditions

Measurement views: Spurious

Group

Sense commands

Syntax

```
[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth}[:RESolution]: AUTO { OFF | ON | 0 | 1 }
[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth}[:RESolution]: AUTO?
```

Arguments

`OFF` or `0` specifies that the resolution bandwidth is set manually using the `[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth}[:RESolution]` command.

`ON` or `1` specifies that the resolution bandwidth is set automatically.

Examples

`SENSe:SPURIOUS:CARRIER:BANDWIDTH:RESOLUTION:AUTO ON` sets the resolution bandwidth automatically.

[SENSe]:SPURious:CARRier:DETection

Sets or queries the carrier detection method. This command is valid when `[SENSe]:SPURious:REFerence` is set to `CARRier`.

Conditions

Measurement views: Spurious

Group

Sense commands

Syntax

```
[SENSe]:SPURious:CARRier:DETection { AVERage | PEAK | AVGLog}
```

[SENSe]:SPURious:CARRier:DETection?

Arguments

AVERage selects the average detection.

PEAK selects the peak detection.

AVGLog selects the average (of logs) detection.

Examples

SENSe:SPURIOUS:CARRIER:DETECTION PEAK selects the peak detection.

[SENSe]:SPURious:CARRier:FREQuency

Sets or queries the carrier frequency in the Spurious measurement. This command is valid when [SENSe]:SPURious:REFerence is set to CARRier.

Conditions

Measurement views: Spurious

Group

Sense commands

Syntax

[SENSe]:SPURious:CARRier:FREQuency <value>
[SENSe]:SPURious:CARRier:FREQuency?

Arguments

<value>::<NRf> specifies the carrier frequency.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples

SENSe:SPURIOUS:CARRIER:FREQUENCY 7.5GHz sets the carrier frequency to 7.5 GHz.

[SENSe]:SPURious:CARRier:THReshold

Sets or queries the threshold level to detect the carrier in the Spurious measurement. This command is valid when [SENSe]:SPURious:REFerence is set to CARRier.

Conditions

Measurement views: Spurious

Group

Sense commands

Syntax

[SENSe]:SPURious:CARRier:THReshold <value>

[SENSe]:SPURious:CARRier:THReshold?

Arguments

<value>::<NRf> specifies the threshold level above which the input signal is determined to be a carrier. Range: -170 to +50 dBm. The unit can be changed by the [SENSe]:POWer:UNITs or UNIT:POWer command.

Examples

SENSe:SPURIOUS:CARRIER:THRESHOLD -25 sets the carrier threshold level to -25 dBm.

[SENSe]:SPURious:CLEar:RESults (No Query Form)

Restarts multi-trace functions (Average and Max Hold).

Conditions

Measurement views: Spurious

Group

Sense commands

Syntax

[SENSe]:SPURious:CLEar:RESults

Arguments

None

Examples

SENSe:SPURIOUS:CLEAR:RESULTS restarts multi-trace functions.

[SENSe]:SPURious[:FREQuency]:OVERlap? (Query Only)

Queries whether any of the frequency ranges (A to T) overlap, including the carrier when [SENSe]:SPURious:REFerence is set to CARRier.

NOTE: If there are any overlaps between the ranges, the measurement will not run correctly.

Conditions

Measurement views: Spurious

Group

Sense commands

Syntax

[SENSe]:SPURious[:FREQuency]:OVERlap?

Arguments

None

Returns

0 (no overlap) or 1 (overlap).

Examples

SENSe:SPURIOUS:FREQUENCY:OVERLAP? might return 1, indicating that some frequency ranges overlap.

[SENSe]:SPURious:LIST

Sets or queries how to list the spurious signals in the Spurious measurement.

Conditions

Measurement views: Spurious

Group

Sense commands

Syntax

```
[SENSe]:SPURious:LIST { ALL | OVERlimit }
[SENSe]:SPURious:LIST?
```

Arguments

ALL lists all of the detected spurious signals.

OVERlimit lists the spurious signals exceeding the limits. Use the [SENSe]:SPURious:RANGe<x>:LIMit command group to set the limits.

Examples

SENSe:SPURIOUS:LIST ALL lists all of the detected spurious signals.

[SENSe]:SPURious:MODE

Sets or queries the frequency range mode in the Spurious measurement.

Conditions

Measurement views: Spurious

Group

Sense commands

Syntax

```
[SENSe]:SPURious:MODE { MULTi | SINGle }
[SENSe]:SPURious:MODE?
```

Arguments

MULTi displays all of the ranges that are enabled. Use the [SENSe]:SPURious:RANGe<x>:STATe command to enable the range.

SINGle displays only the range that the selected spurious signal is in. Use the DISPlay:SPURious:SELect:NUMBer command to select the spurious signal.

Examples

SENSe:SPURIOUS:MODE MULTi displays all of the enabled ranges.

[SENSe]:SPURious:OPTimization

Sets or queries the method of optimizing the gain and input bandwidth in the Spurious measurement.

Conditions

Measurement views: Spurious

Group

Sense commands

Syntax

```
[SENSe]:SPURious:OPTimization { AUTO | MINTime | MAXDynrange| MINNoise }
[SENSe]:SPURious:OPTimization?
```

Arguments

`AUTO` optimizes automatically the gain and input bandwidth.

`MINTime` optimizes the gain and input bandwidth to minimize sweep time.

`MAXDynrange` optimizes the gain and input bandwidth to maximize the dynamic range.

`MINNoise` optimizes the gain and input bandwidth to minimize noise.

Examples

`SENSe:SPURIOUS:OPTIMIZATION AUTO` optimizes automatically the gain and input bandwidth.

[SENSe]:SPURious:POINts:COUNt

Sets or queries the number of sample points on the spectrum trace per range in the Spurious measurement.

Conditions

Measurement views: Spurious

Group

Sense commands

Syntax

```
[SENSe]:SPURious:POINts:COUNt { P801 | P2401 | P4001 | P8001| P10401 }
[SENSe]:SPURious:POINts:COUNt?
```

Arguments

`P801` sets the trace points to 801 per range.

`P2401` sets the trace points to 2401 per range.

`P4001` sets the trace points to 4001 per range.

`P8001` sets the trace points to 8001 per range.

`P10401` sets the trace points to 10401 per range.

Examples

`SENSe:SPURIOUS:POINTS:COUNT P801` sets the trace points to 801 per range.

[SENSe]:SPURious:RANGe<x>:BANDwidth:VIDeo

Sets or queries the video bandwidth (VBW) in the specified frequency range. Programming a specified VBW sets `[SENSe]:SPURious:BANDwidth:VIDeo STATe OFF`.

The parameter <x> = 1 to 20, representing Range A to T, respectively.

Conditions

Measurement views: Spurious

Group

Sense commands

Syntax

```
[SENSe]:SPURious:RANGe<x>:BANDwidth:VIDeo <value>
[SENSe]:SPURious:RANGe<x>:BANDwidth:VIDeo?
```

Related Commands

[SENSe]:SPECtrum:{BANDwidth|BWIDth}:VIDeo:STATe (on page 2383)

Arguments

`<value>::<NRf>` specifies the VBW.

Range: Current RBW/104 (1 Hz minimum) to Current RBW.

Examples

`SENSe:SPURIOUS:RANGE1:BANDWIDTH:VIDEO 200kHz` sets the VBW to 200 kHz for Range A.

[SENSe]:SPURious:RANGe<x>:BANDwidth:VIDeo:STATe

Determines whether to enable or disable the video bandwidth (VBW) in the specified frequency range.

The parameter <x> = 1 to 20, representing Range A to T, respectively.

Conditions

Measurement views: Spurious

Group

Sense commands

Syntax

```
[SENSe]:SPURious:RANGe<x>:BANDwidth:VIDeo:STATe { OFF | ON| 0 | 1 }
[SENSe]:SPURious:RANGe<x>:BANDwidth:VIDeo:STATe?
```

Arguments

`OFF` or `0` disables the VBW.

`ON` or `1` enables the VBW.

Examples

`SENSe:SPURIOUS:RANGE1:BANDWIDTH:VIDEO:STATE ON` enables the VBW for Range A.

[SENSe]:SPURious:RANGe<x>:DETection

Sets or queries the spurious detection method in the specified frequency range. The parameter <x> = 1 to 20, representing Range A to T, respectively.

Conditions

Measurement views: Spurious

Group

Sense commands

Syntax

```
[SENSe]:SPURious:RANGe<x>:DETection { AVERage | PEAK | QUASipeak | CAVerage | CPEak | AVGLog }
[SENSe]:SPURious:RANGe<x>:DETection?
```

Arguments

`AVERage` selects the average detection.

`PEAK` selects the peak detection.

`QUASipeak` selects the quasi-peak detection.

`CAVerage` selects the CISPR average detection.

`CPEak` selects the CISPR peak detection.

`AVGLog` selects the average (of logs) detection.

Examples

`SENSe:SPURIOUS:RANGE1:DETECTION PEAK` selects the peak detection for Range A.

[SENSe]:SPURious:RANGe<x>:EXCursion

Sets or queries the excursion level (how far down the signal must drop between spurious emissions) in the specified frequency range.

The parameter <x> = 1 to 20, representing Range A to T, respectively.

Conditions

Measurement views: Spurious

Group

Sense commands

Syntax

```
[SENSe]:SPURious:RANGe<x>:EXCursion <value>
[SENSe]:SPURious:RANGe<x>:EXCursion?
```

Arguments

`<value>::<NRf>` specifies the excursion level. A signal with amplitude less than the excursion level is considered to be a noise. Range: 1 to 50 dB.

Examples

`SENSe:SPURIOUS:RANGE1:EXCURSION 8` sets the excursion level to 8 dB.

[SENSe]:SPURious:RANGe<x>:FILTer[:SHAPe]

Sets or queries the filter shape to search the specified frequency range for spurious signals.

The parameter <x> = 1 to 20, representing Range A to T, respectively.

Conditions

Measurement views: Spurious

Group

Sense commands

Syntax

```
[SENSe]:SPURious:RANGe<x>:FILTer[:SHAPe] { RBW | MIL6db | CISPr }
[SENSe]:SPURious:RANGe<x>:FILTer[:SHAPe]?
```

Arguments

`RBW` selects the RBW filter.

`MIL6db` selects the -6 dB RBW (MIL) filter.

`CISPr` selects the CISPR filter.

Examples

`SENSe:SPURIOUS:RANGE1:FILTER:SHAPE CISPr` uses the CISPR filter in Range A.

[SENSe]:SPURious:RANGe<x>:FILTer[:SHAPe]:BANDwidth

Sets or queries the filter bandwidth to search the specified frequency range for spurious signals.

The parameter <x> = 1 to 20, representing Range A to T, respectively.

Conditions

Measurement views: Spurious

Group

Sense commands

Syntax

```
[SENSe]:SPURious:RANGe<x>:FILTer[:SHAPe]:BANDwidth <value>
[SENSe]:SPURious:RANGe<x>:FILTer[:SHAPe]:BANDwidth?
```

Arguments

`<value>::<NRf>` specifies the filter bandwidth.

The setting range varies depending on the center frequency and trace detector when using the CISPR filters and detectors. Otherwise, the bandwidth minimum is a function of the span (= (stop frequency) - (start frequency)) of the range and the maximum is fixed at 5 MHz.

By default, the value is set automatically (`[SENSe]:SPURious:RANGe<x>:FILTer[:SHAPe]:BANDwidth:AUTO` is `ON`). When the detection is set to CISPR QPk (`[SENSe]:SPURious:RANGe<x>:DETection` is `QUASipeak`), only one value is allowed, which is set automatically.

Examples

`SENSe:SPURIOUS:RANGE1:FILTER:SHAPE:BANDWIDTH 200kHz` sets the filter bandwidth to 200 kHz for Range A.

[SENSe]:SPURious:RANGe<x>:FILTer[:SHAPe]:BANDwidth:AUTO

Determines whether to set the filter bandwidth automatically or manually for the specified frequency range in the Spurious measurement.

The parameter <x> = 1 to 20, representing Range A to T, respectively.

Conditions

Measurement views: Spurious

Group

Sense commands

Syntax

```
[SENSe]:SPURious:RANGe<x>:FILTer[:SHAPe]:BANDwidth:AUTO { OFF | ON | 0 | 1 }
[SENSe]:SPURious:RANGe<x>:FILTer[:SHAPe]:BANDwidth:AUTO?
```

Arguments

`OFF` or `0` specifies that the filter bandwidth is set manually using the `[SENSe]:SPURious:RANGe<x>:FILTer[:SHAPe]:BANDwidth` command.

`ON` or `1` specifies that the filter bandwidth is set automatically.

Examples

`SENSe:SPURIOUS:RANGE1:FILTER:SHAPE:BANDWIDTH:AUTO ON` sets the filter bandwidth automatically for Range A.

[SENSe]:SPURious:RANGe<x>:FREQuency:STARt

Sets or queries the start frequency of the specified range in the Spurious measurement.

The parameter <x> = 1 to 20, representing Range A to T, respectively.

NOTE: The frequency ranges must not be overlapped. Use the `[SENSe]:SPURious[:FREQuency]:OVERlap?` query to check whether there is any overlap.

Conditions

Measurement views: Spurious

Group

Sense commands

Syntax

```
[SENSe]:SPURious:RANGe<x>:FREQuency:STARt <value>
[SENSe]:SPURious:RANGe<x>:FREQuency:STARt?
```

Arguments

`<value>::<NRf>` specifies the start frequency.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples

`SENSe:SPURIOUS:RANGE1:FREQUENCY:START 1.615GHz` sets the start frequency of Range A to 1.615 GHz.

[SENSe]:SPURious:RANGe<x>:FREQuency:STOP

Sets or queries the stop frequency of the specified range in the Spurious measurement.

The parameter <x> = 1 to 20, representing Range A to T, respectively.

NOTE: The frequency ranges must not be overlapped. Use the `[SENSe]:SPURious[:FREQuency]:OVERlap?` query to check whether there is any overlap.

Conditions

Measurement views: Spurious

Group

Sense commands

Syntax

```
[SENSe]:SPURious:RANGe<x>:FREQuency:STOP <value>
[SENSe]:SPURious:RANGe<x>:FREQuency:STOP?
```

Arguments

`<value>::<NRf>` specifies the stop frequency.

Range:

- RSA5103B – 0 to 3 GHz
- RSA5106B – 0 to 6.2 GHz
- RSA5115B – 0 to 15 GHz
- RSA5126B – 0 to 26.5 GHz

Examples

`SENSe:SPURIOUS:RANGE1:FREQUENCY:STOP 1.715GHz` sets the stop frequency of Range A to 1.715 GHz.

[SENSe]:SPURious:RANGe<x>:LIMit:ABSolute:STARt

Sets or queries the absolute amplitude of the limits at the start (left edge) of the specified range in the Spurious measurement. This command is valid when `[SENSe]:SPURious:RANGe<x>:LIMit:MASK` is set to ABS, AND, or OR.

The parameter <x> = 1 to 20, representing Range A to T, respectively.

Conditions

Measurement views: Spurious

Group

Sense commands

Syntax

```
[SENSe]:SPURious:RANGe<x>:LIMit:ABSolute:STARt <value>
[SENSe]:SPURious:RANGe<x>:LIMit:ABSolute:STARt?
```

Arguments

<value>::<NRf> specifies the absolute start amplitude of the limits.

Range: -170 to +50 dBm.

The unit can be changed by the [SENSe]:POWer:UNITs or UNIT:POWer command.

Examples

SENSe:SPURIOUS:RANGE1:LIMIT:ABSOLUTE:START -30 sets the absolute start amplitude of the limits for Range A to -30 dBm.

[SENSe]:SPURious:RANGe<x>:LIMit:ABSolute:STOP

Sets or queries the absolute amplitude of the limits at the stop (right edge) of the specified range in the Spurious measurement. This command is valid when [SENSe]:SPURious:RANGe<x>:LIMit:MASK is set to ABS, AND, or OR.

The parameter <x> = 1 to 20, representing Range A to T, respectively.

Conditions

Measurement views: Spurious

Group

Sense commands

Syntax

```
[SENSe]:SPURious:RANGe<x>:LIMit:ABSolute:STOP <value>
[SENSe]:SPURious:RANGe<x>:LIMit:ABSolute:STOP?
```

Related Commands

[SENSe]:POWer:UNITs (on page 2269)

UNIT:POWer (on page 2738)

Arguments

<value>::<NRf> specifies the absolute stop amplitude of the limits.

Range: -170 to +50 dBm.

The unit can be changed by the [SENSe]:POWer:UNITs or UNIT:POWer command.

Examples

SENSe:SPURIOUS:RANGE1:LIMIT:ABSOLUTE:STOP -10 sets the absolute stop amplitude of the limits for Range A to -10 dBm.

[SENSe]:SPURious:RANGe<x>:LIMit:MASK

Sets or queries the limit mask function mode for the specified range in the Spurious measurement.

The parameter <x> = 1 to 20, representing Range A to T, respectively.

Conditions

Measurement views: Spurious

Group

Sense commands

Syntax

```
[SENSe]:SPURious:RANGe<x>:LIMit:MASK{ABS|REL|AND|OR|OFF}
[SENSe]:SPURious:RANGe<x>:LIMit:MASK?
```

Arguments

The following table lists the arguments.

Limit mask mode

Argument	Description
ABS	Failure is detected when one of the spurious signals is larger than the absolute amplitude limit.
REL	Failure is detected when one of the spurious signals is larger than the relative amplitude limit.
AND	Failure is detected when one of the spurious signals is larger than the absolute AND relative amplitude limits.
OR	Failure is detected when one of the spurious signals is larger than the absolute OR relative amplitude limit.
OFF	Disables the mask.

To set the absolute amplitude limits, use the `[SENSe]:SPURious:RANGe<x>:LIMit:ABSolute:STARt` and `[SENSe]:SPURious:RANGe<x>:LIMit:ABSolute:STOP` commands.

To set the relative amplitude limits, use the `[SENSe]:SPURious:RANGe<x>:LIMit:RELative:STARt` and `[SENSe]:SPURious:RANGe<x>:LIMit:RELative:STOP` commands.

Examples

`SENSe:SPURIOUS:RANGE1:LIMIT:MASK ABS` specifies that failure is detected when one of the spurious signals is larger than the absolute amplitude limit in Range A.

[SENSe]:SPURious:RANGe<x>:LIMit:RELative:STARt

Sets or queries the relative amplitude of the limits at the start (left edge) of the specified range in the Spurious measurement. This command is valid when `[SENSe]:SPURious:RANGe<x>:LIMit:MASK` is set to REL, AND, or OR.

The parameter <x> = 1 to 20, representing Range A to T, respectively.

Conditions

Measurement views: Spurious

Group

Sense commands

Syntax

```
[SENSe]:SPURious:RANGe<x>:LIMit:RELative:STARt <value>
[SENSe]:SPURious:RANGe<x>:LIMit:RELative:STARt?
```

Arguments

`<value>::<NRf>` specifies the relative start amplitude of the limits.

Range: -100 to 0 dB.

Use the `[SENSe]:SPURious:REFerence` command to select the power reference.

Examples

SENSe:SPURIOUS:RANGE1:LIMIT:RELATIVE:START -30 sets the relative start amplitude of the limits for Range A to -30 dB.

[SENSe]:SPURious:RANGe<x>:LIMit:RELative:STOP

Sets or queries the relative amplitude of the limits at the stop (right edge) of the specified range in the Spurious measurement. This command is valid when [SENSe]:SPURious:RANGe<x>:LIMit:MASK is set to REL, AND, or OR.

The parameter <x> = 1 to 20, representing Range A to T, respectively.

Conditions

Measurement views: Spurious

Group

Sense commands

Syntax

```
[SENSe]:SPURious:RANGe<x>:LIMit:RELative:STOP <value>
[SENSe]:SPURious:RANGe<x>:LIMit:RELative:STOP?
```

Arguments

<value>::<NRf> specifies the relative stop amplitude of the limits.

Range: -100 to 0 dB.

Use the [SENSe]:SPURious:REFerence command to select the power reference.

Examples

SENSe:SPURIOUS:RANGE1:LIMIT:RELATIVE:STOP -10 sets the relative stop amplitude of the limits for Range A to -10 dB.

[SENSe]:SPURious:RANGe<x>:STATe

Determines whether to enable or disable the frequency range in the Spurious measurement.

The parameter <x> = 1 to 20, representing Range A to T, respectively.

Conditions

Measurement views: Spurious

Group

Sense commands

Syntax

```
[SENSe]:SPURious:RANGe<x>:STATe { OFF | ON | 0 | 1 }
[SENSe]:SPURious:RANGe<x>:STATe?
```

Arguments

OFF or 0 disables the frequency range.

ON or 1 enables the frequency range.

Examples

SENSe:SPURIOUS:RANGE1:STATE ON enables Range A.

[SENSe]:SPURious:RANGe<x>:THReshold

Sets or queries the threshold level to detect spurious signals in the specified frequency range.

The parameter <x> = 1 to 20, representing Range A to T, respectively.

Conditions

Measurement views: Spurious

Group

Sense commands

Syntax

```
[SENSe]:SPURious:RANGe<x>:THReshold <value>
[SENSe]:SPURious:RANGe<x>:THReshold?
```

Arguments

<value>::<NRf> specifies the threshold level above which the signal is determined to be spurious. Range: -50 to +30 dBm.

The unit can be changed by the [SENSe]:POWer:UNITs or UNIT:POWer command.

Examples

SENSe:SPURIOUS:RANGE1:THRESHOLD -25 sets the threshold level to -25 dBm in Range A.

[SENSe]:SPURious:REFerence

Sets or queries the power reference in the Spurious measurement.

Conditions

Measurement views: Spurious

Group

Sense commands

Syntax

```
[SENSe]:SPURious:REFerence { CARRier | MANual | NONE }
[SENSe]:SPURious:REFerence?
```

Arguments

CARRier uses the carrier as the power reference.

MANual sets the power reference using the [SENSe]:SPURious:REFerence:MANual:POWer command.

NONE uses no reference.

Examples

SENSe:SPURIOUS:REFERENCE CARRier uses the carrier as the power reference.

[SENSe]:SPURious:REFerence:MANual:POWer

Sets or queries the reference power level in the Spurious measurement. This command is valid when `[SENSe]:SPURious:REFerence` is set to MANual.

Conditions

Measurement views: Spurious

Group

Sense commands

Syntax

```
[SENSe]:SPURious:REFerence:MANual:POWer <value>
[SENSe]:SPURious:REFerence:MANual:POWer?
```

Arguments

`<value>::<NRf>` specifies the reference power level. Range: -170 to +50 dBm.

The unit can be changed by the `[SENSe]:POWer:UNITs` or `UNIT:POWer` command.

Examples

`SENSe:SPURIOUS:REFERENCE:MANUAL:POWER -25` sets the reference power level to -25 dBm.

[SENSe]:SQUality[:MEASview<y>]:EVM:ALL[:STATe]

Sets or queries whether or not the display of both EVM and Offset EVM results is enabled. Offset EVM appears only for OQPSK and SOQPSK modulation types.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Sense commands

Syntax

```
[SENSe]:SQUality[:MEASview<y>]:EVM:ALL[:STATe]
[SENSe]:SQUality[:MEASview<y>]:EVM:ALL[:STATe]?
```

Arguments

`OFF` or 0 disables the local lockout operation.

`ON` or 1 enables the local lockout operation.

[SENSe]:SQUality[:MEASview<y>]:SELEcted:SOURce

Sets or queries the Source of Signal quality for a selected Measurement view number.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal Quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Source cannot be changed when acquisition or analysis is in progress. Stop the acquisition and change the Source.

Group

Sense commands

Syntax

```
[SENSe]:SQUality[:MEASview<y>]:SELEcted:SOURce {"Source 1" | "Source 2" | "Source 3" | "Source 4"}
[SENSe]:SQUality[:MEASview<y>]:SELEcted:SOURce?
```

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1.

Examples

`SENSe:SQUality:MEASview1:SELECTED:SOURce?` queries the Source for Signal quality Measurement View 1.

[SENSe]:STABle[:MEASview<y>]:SELEcted:SOURce

Sets or queries the Source of Symbol table for a selected Measurement view number.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Symbol table

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod symbol table).

Source cannot be changed when the acquisition or analysis is in progress. Stop the acquisition and change the Source.

Group

Sense commands

Syntax

```
[SENSe]:STABle [:MEASview<y>]:SELEcted:SOURce {"Source 1" | "Source 2" | "Source 3" | "Source 4"}
[SENSe]:STABle [:MEASview<y>]:SELEcted:SOURce?
```

Arguments

The parameter <y>=1 to 8 represents measurement index based on the number of sources of connected instrument, where MEASview keyword is optional by default and y is 1.

Examples

`SENSe:STABle:MEASview1:SELEcted:SOURce?` queries the Source for Symbol table Measurement View 1.

[SENSe]:SIGNalvu:ACQuisition:CONTrol:TRIGger

Sets or queries the Trigger Settings checkbox state.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
[SENSe]:SIGNalvu:ACQuisition:CONTrol:TRIGger { OFF | ON | 0 | 1 }
[SENSe]:SIGNalvu:ACQuisition:CONTrol:TRIGger?
```

Arguments

`OFF` or `0` disables the Trigger Settings.

`ON` or `1` enables the Trigger Settings.

Examples

`SENSe:SIGNalvu:ACQuisition:CONTrol:TRIGger ON` enables the Trigger setting.

[SENSe]:SIGNalvu:ACQuisition:CONTrol:CHANnel

Sets or queries the Channel Settings checkbox state.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:SIGNalvu :ACQuisition:CONTrol:CHANnel { OFF | ON | 0 | 1 }
[SENSe]:SIGNalvu :ACQuisition:CONTrol:CHANnel?
```

Arguments

`OFF` or `0` disables the channel settings.

`ON` or `1` enables the channel settings.

Examples

`SENSe:SIGNalvu :ACQuisition:CONTrol:CHANnel ON` enables the Channel Settings.

[SENSe]:SIGNalvu:ACQuisition:FFRame:ACTual? (Query only)

Queries the actual number of Fast Frames.

Conditions

Requires an RSA/MSO be connected with Signalvu. Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:SIGNalvu:ACQuisition:FFRame:ACTual?
```

Arguments

None

Returns

<NR1> The actual number of Fast Frames.

Examples

SENSe:SIGNalvu :ACQuisition:FFRame:ACTual? might return 178, indicating that the actual number of Fast Frames is 178.

[SENSe]:SIGNalvu:ACQuisition:FFRame:LIMit

Sets or queries the limit number of Fast Frames.

Conditions

Requires an RSA/MSO be connected with SIGNalvu. Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:SIGNalvu:ACQuisition:FFRame:LIMit
[SENSe]:SIGNalvu:ACQuisition:FFRame:LIMit?
```

Arguments

<value>::<NR1> specifies the fast frame limit. Range: 1 to 65535 frames.

Where

1 to 65535 frames for RSA 5000A series. For others, the range is 1 to 220 -1 frames.

Examples

SENSe:SIGNalvu :ACQUISITION:FFRAME:LIMIT 500 sets the Fast Frame limit to 500.

[SENSe]:SIGNalvu:ACQuisition:FFRame:STATe

Determines whether to enable or disable the Fast Frame.

Conditions

Requires an RSA/MSO be connected with SIGNalvu. Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:SIGNalvu:ACQuisition:FFRame:STATe { OFF | ON | 0 | 1 }
[SENSe]:SIGNalvu:ACQuisition:FFRame:STATe?
```

Arguments

OFF or 0 disables the Fast Frame.

ON or 1 enables the Fast Frame.

Examples

`SENSe:SIGNalvu :ACQUISITION:FFRAME:STATE ON` enables the Fast Frame.

[SENSe]:TDIagram[:MEASview<y>]:SELEcted:SOURce

Sets or queries the Source of Trellis diagram for a selected Measurement view number.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Trellis diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Trellis diagram).

Source cannot be changed when the acquisition or analysis is in progress. Stop the acquisition and change the Source.

Group

Sense commands

Syntax

```
[SENSe]:TDIagram [:MEASview<y>]:SELEcted:SOURce {"Source 1" | "Source 2" | "Source 3" | "Source 4"}
[SENSe]:TDIagram [:MEASview<y>]:SELEcted:SOURce?
```

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1.

Examples

`SENSe:TDIagram:MEASview1:SELEcted:SOURce?` queries the Source for Trellis diagram Measurement View 1.

SENSe:TOFF:POWEr:SPECtral:DENSity

Sets or queries the limit set for comparison of the Toff power spectral density value in the Limits tab of the LTE Power vs Time display settings control panel.

Conditions

Measurement view: LTE Power vs Time

Group

Sense commands

```
Syntax
SENSe:TOFF:POWEr:SPECtral:DENSity <value>
SENSe:TOFF:POWEr:SPECtral:DENSity?
```

Arguments

`<value>::<NRf>` specifies the Toff power spectral density value.

Returns

`<NRf>`

Examples

SENSe:TOFF:POWER:SPECTRAL:DENSITY? might return -83.000000000, indicating that the Toff power spectral density is set to -83 dBm.

[SENSe]:TOVerview[:MEASview<y>]:FREQuency:CENTer

Sets or queries the center frequency in the time overview.

Conditions

Measurement views: Time overview

Group

Sense commands

Syntax

```
[SENSe]:TOVerview[:MEASview<y>]:FREQuency:CENTer <value>
[SENSe]:TOVerview[:MEASview<y>]:FREQuency:CENTer?
```

Arguments

<value>::<NRf> specifies the center frequency.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Returns

<NRf>

Examples

SENSe:TOVERVIEW:MEASVIEW1:FREQUENCY:CENTER 7.5GHz sets the center frequency to 7.5 GHz.

[SENSe]:TOVerview[:MEASview<y>]:MAXTracepoints

Sets or queries the maximum trace points in the time overview.

Conditions

Measurement views: Time overview

Group

Sense commands

Syntax

```
[SENSe]:TOVerview[:MEASview<y>]:MAXTracepoints { ONEK | TENK | HUNDredk| NEVerdecimate }
[SENSe]:TOVerview[:MEASview<y>]:MAXTracepoints?
```

Arguments

ONEK sets the maximum trace points to 1 k.

TENK sets the maximum trace points to 10 k.

HUNDredk sets the maximum trace points to 100 k.

NEVerdecimate never decimates the trace points.

Returns

<NRf>

Examples

SENSe:TOVERVIEW:MEASVIEW1:MAXTRACEPOINTS TENK sets the maximum trace points to 10 k.

[SENSe]:TOVerview[:MEASview<y>]:TIME:PER:DIVision

This command sets or queries the time per division (dB/div) value in the Time Overview display.

Conditions

Measurement view: Time Overview

Group

Sense commands

Syntax

```
[SENSe]:TOVerview[:MEASview<y>]:TIME:PER:DIVision <NRf>
[SENSe]:TOVerview[:MEASview<y>]:TIME:PER:DIVision?
```

Arguments

<NRf> is the dB/div value.

Range: .01 to 20

Returns

<NR2> = floating point value without an exponent, which is the time per division value.

Examples

SENSe:TOVerview:MEASVIEW1:TIME:PER:DIV might return 10.0000000000, indicating that the time per division value is set to 10.0 dB.

[SENSe]:TOVerview[:MEASview<y>]:SELEcted:SOURce

Sets or queries the Measurement view number for the Time Overview Display.

Conditions

Measurement view: Time overview with specified Measurement view number

Source cannot be changed when acquisition or analysis is in progress. Stop the acquisition and change the Source.

Group

Sense commands

Syntax

```
[SENSe]:TOVerview[:MEASview<y>]:SELEcted:SOURce {"Source 1" | "Source 2" | "Source 3" | "Source 4"}
[SENSe]:TOVerview[:MEASview<y>]:SELEcted:SOURce?
```

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

<NRf>

Examples

SENSe:TOVERVIEW:MEASVIEW1:SELECTED:SOURCE? gives the Measurement view number for the Time overview source.

[SENSe]:TRACe:SPURious:COUNt:RESet (No Query Form)

Clears the multi-function (Max Hold or Average) data and counter and restarts the process in the Spurious view.

Conditions

Measurement view: Spurious Display

Group

Sense commands

Syntax

[SENSe]:TRACe:SPURious:COUNt:RESet

Arguments

None

Returns

None

Examples

SENSe:TRACE:SPURIOUS:COUNT:RESET clears the multi-function data and counter and restarts the process.

[SENSe]:TRACe<x>:SPURious:COUNt

Sets and queries the count value for the selected trace in the Spurious measurement.

Conditions

Measurement view: Spurious Display

Group

Sense commands

Syntax

[SENSe]:TRACe<x>:SPURious:COUNt <value>
[SENSe]:TRACe<x>:SPURious:COUNt

Arguments

<x> = 1,2,3,4 is the Trace

<value> = < NR1> Range is 1 to 10000

Returns

<NR1>

Examples

SENSe:TRACe4:SPURious:COUNt 60 sets the count value to 60 for Trace 4.

[SENSe]:TRACe<x>:SPURious:COUNt:ENABle

Determines whether to enable or disable the count check box for the specified trace in the Spurious measurement.

Conditions

Measurement view: Spurious Display

Group

Sense commands

Syntax

```
[SENSe]:TRACe<x>:SPURious:COUNt:ENABle {0 | OFF | 1 | ON}
[SENSe]:TRACe<x>:SPURious:COUNt:ENABle?
```

Arguments

<x> = 1,2,3,4 is the Trace

ON or 1 to enable the count check box

OFF or 0 to disable the count check box

Returns

<Boolean> 0, 1

Examples

SENSe:TRACe2:SPURious:COUNt:ENABle ON enables the count check box for Trace 2.

[SENSe]:TRACe<x>:SPURious:ENABle

Determines whether to enable or disable show check box for the specified trace in the Spurious measurement.

Conditions

Measurement view: Spurious Display

Group

Sense commands

Syntax

```
[SENSe]:TRACe<x>:SPURious:ENABle {0 | OFF | 1 | ON}
[SENSe]:TRACe<x>:SPURious:ENABle?
```

Arguments

<x> = 1,2,3,4 is the Trace

ON or 1 to enable the show check box

OFF or 0 to disable the show check box

Returns

<Boolean> 0, 1

Examples

SENSe:TRACe2:SPURious:ENABle 1 enables the show check box for Trace 2.

[SENSe]:TRACe<x>:SPURious:FREeze

Determines whether to enable or disable the freeze check box for the specified trace in the Spurious measurement.

Conditions

Measurement view: Spurious Display

Group

Sense commands

Syntax

```
[SENSe]:TRACe<x>:SPURious:FREeze {0 | OFF | 1 | ON}
[SENSe]:TRACe<x>:SPURious:FREeze?
```

Arguments

<x> = 1,2,3,4 is the Trace

ON or 1 to enable the freeze check box

OFF or 0 to disable the freeze check box

Returns

<Boolean> 0, 1

Examples

SENSe:TRACe3:SPURious:FREeze ON enables the freeze check box for Trace 3.

[SENSe]:TRACe<x>:SPURious:FUNCtion

Sets and queries the required function for the selected trace in Spurious measurement.

Conditions

Measurement view: Spurious Measurement Display

Group

Sense commands

Syntax

```
[SENSe]:TRACe<x>:SPURious:FUNCtion {NONE | MAXH | AVER | AVGL}
[SENSe]:TRACe<x>:SPURious:FUNCtion?
```

Arguments

<x> = 1,2,3,4 is the Trace

"NONE" to select Normal

"MAXH" to select Max Hold

"AVER" to select Avg (VRMS)

"AVGL" to select Avg (of logs)

Returns

<String> NONE or MAXH or AVER or AVGL

Examples

SENSe:TRACe1:SPURious:FUNCtion NONE sets the function to Normal for Trace 1.

[SENSe]:TRACe<x>:SPURious:SELect

Sets the specified trace as the active trace in the Spurious display. The query returns the active trace.

Conditions

Measurement view: Spurious Display

Group

Sense commands

Syntax

```
[SENSe]:TRACe<x>:SPURious:SELect
[SENSe]:TRACe<x>:SPURious:SELect?
```

Arguments

<x> = 1,2,3,4 is the Trace

Returns

<Boolean> 0, 1

Examples

SENSe:TRACe2:SPURious:SELect sets Trace 2 as the active trace.

SENSe:TXGain:{BANDwidth|BWIDth}[:RESolution]

Sets or queries the resolution bandwidth.

Conditions

Measurement view: Transmission Gain

Group

Sense commands

Syntax

```
SENSe:TXGain:{BANDwidth|BWIDth}[:RESolution] <value>
SENSe:TXGain:{BANDwidth|BWIDth}[:RESolution]?
```

Arguments

<value>::=<NRf> specifies the RBW. Range: 1 kHz to 1 MHz.

Returns

See Arguments.

Examples

`SENSe:TXGain:BANDwidth:RESolution 200E3` sets the RBW to 200 kHz.

SENSe:TXGain:FREQuency:CENTer

Sets or queries the frequency at the center of the selected Span.

Conditions

Measurement view: Transmission Gain

Group

Sense commands

Syntax

```
SENSe:TXGain:FREQuency:CENTer <value>
SENSe:TXGain:FREQuency:CENTer?
```

Arguments

`<value>::=<NRf>` specifies the center frequency.

Range: 1 kHz to 1 MHz.

Returns

See Arguments.

Examples

`SENSe:TXGAIN:FREQUENCY:CENTER 1E9` sets the center frequency to 1 GHz.

SENSe:TXGain:FREQuency:POINts

Sets or queries the number of frequency steps in the measurement range.

Conditions

Measurement view: Transmission Gain

Group

Sense commands

Syntax

```
SENSe:TXGain:FREQuency:POINts <value>
SENSe:TXGain:FREQuency:POINts?
```

Arguments

`<value>::=<NR1>` specifies number of frequency steps.

Returns

See Arguments.

Examples

`SENSe:TXGAIN:FREQUENCY:POINTS 101` sets the number of frequency steps to 101.

SENSe:TXGain:FREQuency:SPAN

Sets or queries the difference between the start and stop frequencies (span).

Conditions

Measurement view: Transmission Gain

Group

Sense commands

Syntax

```
SENSe:TXGain:FREQuency:SPAN <value>
SENSe:TXGain:FREQuency:SPAN?
```

Arguments

`<value>::=<NR1>` specifies the frequency span.

Returns

See Arguments.

Examples

`SENSe:TXGAIN:FREQUENCY:SPAN 100E6` sets the frequency span to 100 MHz.

SENSe:TXGain:FREQuency:STARt

Sets or queries the lowest frequency in the span.

Conditions

Measurement view: Transmission Gain

Group

Sense commands

Syntax

```
SENSe:TXGain:FREQuency:STARt <value>
SENSe:TXGain:FREQuency:STARt?
```

Related Commands

SENSe:TXGain:FREQuency:STOP (on page 2424)

Arguments

`<value>::=<NR1>` specifies the lowest frequency in the span.

Returns

See Arguments.

Examples

`SENSe:TXGAIN:FREQUENCY:START 1E9` sets the lowest frequency to 1 GHz.

SENSe:TXGain:FREQuency:STEP

Sets or queries the frequency increment. Measurements are made at frequency points from Start Freq to Stop Freq in Step Size increments.

Conditions

Measurement view: Transmission Gain

Group

Sense commands

Syntax

```
SENSe:TXGain:FREQuency:STEP <value>
SENSe:TXGain:FREQuency:STEP?
```

Arguments

`<value>::=<NRf>` specifies the frequency increment.

Returns

See Arguments.

Examples

`SENSe:TXGAIN:FREQUENCY:STEP 1E6` sets the frequency increment to 1 MHz.

SENSe:TXGain:FREQuency:STOP

Sets or queries the highest frequency in the span.

Conditions

Measurement view: Transmission Gain

Group

Sense commands

Syntax

```
SENSe:TXGain:FREQuency:STOP <value>
SENSe:TXGain:FREQuency:STOP?
```

Related Commands

SENSe:TXGain:FREQuency:STARt (on page 2423)

Arguments

`<value>::=<NRf>` specifies the highest frequency.

Returns

See Arguments.

Examples

`SENSe:TXGAIN:FREQUENCY:STOP 2E9` sets the highest frequency to 2 GHz.

[SENSe]:USETtings (No Query Form)

Updates the analyzer settings. This command is useful when you need to set the analyzer including the RF attenuation before taking data acquisition. Unless this command is executed, the attenuation value is not set until acquisition is taken.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:USETtings
```

Arguments

None

Examples

`SENSe:USETTINGS` updates settings.

[SENSe]:WLAN:AD:ANALysis:BLOCkwise:CORRection:STATe

Enables or disables the Enable Blockwise Correction check box in the Adv Params tab of the Settings control panel for the 802.11ad displays.

Conditions

Measurement views: All 802.11ad measurement views

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AD:ANALysis:BLOCkwise:CORRection:STATe { 0 | OFF | 1 | ON }
[SENSe]:WLAN:AD:ANALysis:BLOCkwise:CORRection:STATe?
```

Arguments

`0` or `OFF` disables Blockwise Correction (unchecks check box)

`1` or `ON` enables Blockwise Correction (checks check box)

Returns

`<Boolean>` 0,1

Examples

`SENSe:WLAN:AD:ANALYSIS:BLOCKWISE:CORRECTION:STATE 1` enables Blockwise Correction.

[SENSe]:WLAN:AD:ANALysis:DC:OFFSet:CORRection:STATe

Enables or disables the Enable DC Offset Correction check box for all 802.11ad measurement views (set in the Advanced Params tab of the Settings control panel).

Conditions

Measurement views: All 802.11ad measurement views

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AD:ANALysis:DC:OFFSet:CORRection:STATe {0 | OFF| 1 | ON }
[SENSe]:WLAN:AD:ANALysis:DC:OFFSet:CORRection:STATe?
```

Arguments

`0` or `OFF` disables Enable DC Offset Correction (unchecks check box)

`1` or `ON` enables Enable DC Offset Correction (checks check box)

Returns

`<Boolean>` 0,1

Examples

`SENSe:WLAN:AD:ANALYSIS:DC:OFFSET:CORRECTION:STATE 0` disables the Enable DC Offset Correction check box.

[SENSe]:WLAN:AD:ANALysis:EQUAlizer:CONVergence

Sets or queries the Convergence value (set in the Advanced Params tab of the Settings control panel) for the 802.11ad displays.

(If you have SignalVu for scopes, this command is only available for SignalVu version 3.10.0030 and above.)

Conditions

Measurement views: All 802.11ad measurement views.

It is applicable only when ADAPtive is set as the Adaptive Equalizer.

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AD:ANALysis:EQUAlizer:CONVergence<value>
[SENSe]:WLAN:AD:ANALysis:EQUAlizer:CONVergence?
```

Arguments

`<value>=<NRf>`

Returns

`<NRf>`

Examples

SENSe:WLAN:AD:ANALYSIS:EQUALIZER:CONVERGENCE? might return 2.000E-3, indicating the Convergence value is 2.000E-3.

[SENSe]:WLAN:AD:ANALysis:EQUAlizer:LENGth

Sets or queries the adaptive equalizer Length (set in the Advanced Params tab of the Settings control panel) for the 802.11ad displays.

(If you have SignalVu for scopes, this command is only available for SignalVu version 3.10.0030 and above.)

Conditions

Measurement views: All 802.11ad measurement views

Only applies when ADAPtive is set as the Adaptive Equalizer

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AD:ANALysis:EQUAlizer:LENGth <value>
[SENSe]:WLAN:AD:ANALysis:EQUAlizer:LENGth?
```

Arguments

<value>=<NR1>

Returns

<NR1>

Examples

SENSe:WLAN:AD:ANALYSIS:EQUALIZER:LENGTH 25 will set the Length to 25.

[SENSe]:WLAN:AD:ANALysis:EQUAlizer:MODE

Sets or queries the equalizer mode (set in the Advanced Params tab of the Settings control panel) for the 802.11ad displays.

(If you have SignalVu for scopes, this command is only available for SignalVu version 3.10.0030 and above.)

Conditions

Measurement views: All 802.11ad measurement views

It is applicable only when ADAPtive is set as the Adaptive Equalizer.

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AD:ANALysis:EQUAlizer:MODE {TRA | HOLD}
[SENSe]:WLAN:AD:ANALysis:EQUAlizer:MODE?
```

Arguments

TRA will set the mode to Train

HOLD will set the mode to Hold

Returns

<String> = TRA or HOLD

Examples

SENSe:WLAN:AD:ANALYSIS:EQUALIZER:MODE TRA sets the equalizer mode to Train.

[SENSe]:WLAN:AD:ANALysis:EQUAlizer:RESet (No Query Form)

Resets the equalization value (button in the Advanced Params tab of the Settings control panel) for the 802.11ad displays.

(If you have SignalVu for scopes, this command is only available for SignalVu version 3.10.0030 and above.)

Conditions

Measurement views: All 802.11ad measurement views

Only applies when ADAPtive is set as the Adaptive Equalizer

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AD:ANALysis:EQUAlizer:RESet
```

Arguments

None

Returns

None

Examples

SENSe:WLAN:AD:ANALYSIS:EQUALIZER:RESET resets the equalization value.

[SENSe]:WLAN:AD:ANALysis:EQUAlizer:TAPS

Sets or queries the Taps value in the Advanced Params tab of the Settings control panel for the 802.11ad displays.

(If you have SignalVu for scopes, this command is only available for SignalVu version 3.10.0030 and above.)

Conditions

Measurement views: All 802.11ad measurement views

Only applies when ADAPtive is set as the Adaptive Equalizer

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AD:ANALysis:EQUAlizer:TAPS <value>
[SENSe]:WLAN:AD:ANALysis:EQUAlizer:TAPS?
```

Arguments

<value>=<NR1>

Returns

<NR1>

Examples

SENSe:WLAN:AD:ANALYSIS:EQUALIZER:TAPS 56 sets the Tap value to 56.

[SENSe]:WLAN:AD:ANALysis:EQUAlizer:TSRatio

Sets or queries the Taps/Symbols value (set in the Advanced Params tab of the Settings control panel) for the 802.11ad displays.

(If you have SignalVu for scopes, this command is only available for SignalVu version 3.10.0030 and above.)

Conditions

Measurement views: All 802.11ad measurement views

It is applicable only when ADAPtive is set as the Adaptive Equalizer

Group

Sense commands

Syntax

[SENSe]:WLAN:AD:ANALysis:EQUAlizer:TSRatio { ONE | TWO | FOUR }

[SENSe]:WLAN:AD:ANALysis:EQUAlizer:TSRatio?

Arguments

ONE, TWO, FOUR

Returns

<String> = either ONE, TWO, or FOUR

Examples

SENSe:WLAN:AD:ANALYSIS:EQUALIZER:TSRATIO might return FOUR, indicating the Taps/Symbols value is 4.

[SENSe]:WLAN:AD:ANALysis:EQUAlizer:TYPE

It sets or Queries the Equalizer to OFF, Non Adaptive Equalizer and Adaptive Equalizer in the settings control panel common to all the WLAN 802.11ad displays.

Conditions

Measurement views: All WLAN 802.11ad measurement views

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AD:ANALysis:EQUAlizer:TYPE {NONE | NONA | ADAptive}
[SENSe]:WLAN:AD:ANALysis:EQUAlizer:TYPE?
```

Arguments

NONE will set the Equalizer to OFF.

NONA will set the Equalizer to Non Adaptive Equalizer.

ADAptive will set the Equalizer to Adaptive Equalizer.

Examples

SENSe:WLAN:AD:ANALYSIS:EQUALIZER:TYPE NONA will set the Equalizer to Non Adaptive Equalizer.

[SENSe]:WLAN:AD:ANALysis:EQUAlizer:TYPE

Sets or queries the Equalizer type in the 802.11ad measurement views.

(If you have SignalVu for scopes, this command is only available for SignalVu version 3.10.0030 and above.)

Conditions

Measurement views: All 802.11ad measurement views

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AD:ANALysis:EQUAlizer:TYPE { NONE | NONA | ADAptive }
[SENSe]:WLAN:AD:ANALysis:EQUAlizer:TYPE?
```

Arguments

NONE will set the Equalizer to OFF

NONA will set the Equalizer to Non Adaptive Equalizer

ADAPtive will set the Equalizer to Adaptive Equalizer

Returns

<String> NONE or NONA or ADAPtive

Examples

SENSe:WLAN:AD:ANALYSIS:EQUALIZER:TYPE ADAptive will set the equalizer to Adaptive Equalizer.

[SENSe]:WLAN:AD:ANALysis:HANDle:MULTi:PATH:CHANnel:STATe

By enabling this flag, severe multipath can be equalized. By default this flag is set to 0.

Conditions

Measurement views: All WLAN 802.11ad measurement views

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AD:ANALysis:HANDle:MULTi:PATH:CHANnel:STATe {0| 1}
[SENSe]:WLAN:AD:ANALysis:HANDle:MULTi:PATH:CHANnel:STATe?
```

Arguments

0 severe multipath cannot be equalized (checkbox is unchecked)

1 severe multipath can be equalized (checkbox is checked)

Returns

0 or 1, <NRf>

Examples

SENSe:WLAN:AD:ANALYSIS:HANDLE:MULTI:PATH:CHANNEL:STATE 1 equalizes the severe multipath.

[SENSe]:WLAN:AD:ANALysis:LENGth

Specifies the length of the analysis period to be used in all the 802.11ad measurement views (set in the Analysis Time tab of the Settings control panel). The analysis length is specified in either symbols or seconds, depending on the Units setting.

Conditions

Measurement views: All 802.11ad measurement views

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AD:ANALysis:LENGth <value>
[SENSe]:WLAN:AD:ANALysis:LENGth?
```

Arguments

<value>=<NRf>

Returns

<NRf> = analysis length. The minimum value depends on the modulation type

Examples

SENSe:WLAN:AD:ANALYSIS:LENGTH 24.7835200000E-6 sets the analysis length to 24.784 µs.

[SENSe]:WLAN:AD:ANALysis:LENGth:ACTual? (Query Only)

Returns the value of the actual analysis length for the 802.11ad measurement views (in the Analysis Time tab of the Settings control panel).

Conditions

Measurement views: All 802.11ad measurement views

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AD:ANALysis:LENGth:ACTual?
```

Arguments

None

Returns

<NRf> = actual analysis length in seconds

Examples

SENSe:WLAN:AD:ANALYSIS:LENGTH:ACTUAL? might return 2.840E-6, indicating that the actual analysis length is 2.840 µs.

[SENSe]:WLAN:AD:ANALysis:LENGth:AUTO

Enables or disables the Auto checkbox state for analysis length in all the 802.11ad displays (set in the Analysis Time tab of the Settings control panel). This setting determines if the analysis length display is set automatically (enabled) or manually (disabled).

Conditions

Measurement views: All 802.11ad measurement views

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AD:ANALysis:LENGth:AUTO {0 | OFF | 1 | ON}
[SENSe]:WLAN:AD:ANALysis:LENGth:AUTO?
```

Arguments

0 or OFF sets the analysis length to manual (box is unchecked)

1 or ON sets the analysis length to automatic (box is checked)

Returns

<Boolean> 0,1

Examples

SENSe:WLAN:AD:ANALYSIS:LENGTH:AUTO ON sets the analysis length value automatically based on the requirements of the selected display.

[SENSe]:WLAN:AD:ANALysis:OFFSet

Sets or queries the analysis offset value in all the 802.11ad measurement displays (set in the Analysis Time tab in the Settings control panel). The query command returns the result in symbols or seconds, depending on the unit choice selected in the Analysis Time tab of the Settings control panel.

Conditions

Measurement views: All 802.11ad measurement views

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AD:ANALysis:OFFSet <value>
[SENSe]:WLAN:AD:ANALysis:OFFSet?
```

Arguments

<value>=<NRf>

Returns

<NRf> = analysis offset

Examples

SENSe:WLAN:AD:ANALYSIS:OFFSET -1.2E-6 sets the analysis offset to -1.200 µs.

[SENSe]:WLAN:AD:ANALysis:OFFSet:AUTO

Enables or disables the Auto check box for analysis offset in all the 802.11ad measurement views (set in the Analysis Time tab of the Settings control panel). This setting determines if the analysis offset displays is set automatically (enabled) or manually (disabled).

Conditions

Measurement views: All 802.11ad measurement views

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AD:ANALysis:OFFSet:AUTO {0 | OFF | 1 | ON}
[SENSe]:WLAN:AD:ANALysis:OFFSet:AUTO?
```

Arguments

0 or OFF sets the analysis offset to disabled (box is unchecked)

1 or ON sets the analysis offset to enabled (box is checked)

Returns

<Boolean> 0,1

Examples

SENSe:WLAN:AD:ANALYSIS:OFFSET:AUTO 1 sets the analysis offset to automatic.

[SENSe]:WLAN:AD:ANALysis:SWAP:IQ

Sets or queries whether or not to swap I and Q data.

Conditions

Measurement views: All WLAN 802.11ad measurement views

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AD:ANALysis:SWAP:IQ {0 | 1}
[SENSe]:WLAN:AD:ANALysis:SWAP:IQ?
```

Arguments

0 does not swap I and Q data (checkbox is unchecked.)

1 swaps I and Q data (checkbox is checked.)

Returns

0 or 1, <NRf>

Examples

SENSe:WLAN:AD:ANALYSIS:SWAP:IQ 1 swaps I and Q data.

[SENSe]:WLAN:AD:ANALysis:TIMe:UNITs

Sets or queries the analysis time units for all 802.11ad measurement views (set in the Analysis Time tab of the Settings control panel).

Conditions

Measurement views: All 802.11ad measurement views

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AD:ANALysis:TIMe:UNITs {SEC, SYMB}
[SENSe]:WLAN:AD:ANALysis:TIMe:UNITs?
```

Arguments

SEC sets the time units to seconds

SYMB sets the time units to symbols

Returns

See Arguments

Examples

SENSe:WLAN:AD:ANALYSIS:TIME:UNITS SEC sets the analysis time units to seconds.

[SENSe]:WLAN:AD(:BANDwidth|:BWIDth):MODE

Sets or queries the measurement BW mode in all the 802.11ad measurement displays (set in the Analysis Params tab in the Settings control panel).

Conditions

Measurement views: All 802.11ad measurement views

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AD(:BANDwidth|:BWIDth):MODE { MAN | AUTO | LSP}
[SENSe]:WLAN:AD(:BANDwidth|:BWIDth):MODE?
```

Arguments

MAN sets the mode to Manual

AUTO sets the mode to Auto

LSP sets the mode to Link to Span

Returns

See Arguments

Examples

SENSe:WLAN:AD(:BANDWIDTH|:BWIDTH):MODE LSP will set the Measurement BW mode to Link to Span.

[SENSe]:WLAN:AD(:BANDwidth|:BWIDth):TINTerval

Sets or queries the Measurement BW in all the 802.11ad measurement displays (set in the Analysis Params tab in the Settings control panel).

Conditions

Measurement views: All 802.11ad measurement views

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AD(:BANDwidth|:BWIDth):TINTerval <value>
[SENSe]:WLAN:AD(:BANDwidth|:BWIDth):TINTerval?
```

Arguments

<value>=<NRf>

Returns

<NRf> = Measurement BW value in Hz

Examples

SENSe:WLAN:AD:BANDWIDTH:TINTERVAL 5 will set the Measurement BW to 5 Hz.

[SENSe]:WLAN:AD:EVM:BURSt:COUNt

Sets or queries the total number of bursts to be analyzed (set in the EVM tab of the Settings control panel) for the display.

Conditions

Measurement views: 802.11ad Constellation, EVM vs Time, and Summary view

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AD:EVM:BURSt:COUNt <value>
[SENSe]:WLAN:AD:EVM:BURSt:COUNt?
```

Arguments

<value>=<NR1> Range 1 to 99

Returns

<NR1> = number of bursts

Examples

SENSe:WLAN:AD:EVM:BURST:COUNT? might return 20, indicating the total number of bursts to be analyzed is 20.

[SENSe]:WLAN:AD:EVM:BURSt:COUNt:ENABle

Enables or disables the Max Bursts to Avg check box in the EVM tab of the Settings control panel for the display.

Conditions

Measurement views: 802.11ad Constellation, EVM vs Time, and Summary view

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AD:EVM:BURSt:COUNt:ENABle {0 | OFF | 1 | ON }
[SENSe]:WLAN:AD:EVM:BURSt:COUNt:ENABle?
```

Arguments

0 or OFF disables the Max Bursts to Avg (unchecks check box)

1 or ON enables the Max Bursts to Avg (checks check box)

Returns

<Boolean> 0,1

Examples

SENSe:WLAN:AD:EVM:BURST:COUNT:ENABLE? might return 1, indicating the Max Bursts to Avg option is enabled.

[SENSe]:WLAN:AD:EVM:DATA:ANALyze:ALL:SYMBols

Enables or disables the Analyze All check box in the EVM tab of the Settings control panel for the display. When the Analyze All check box is enabled, all the data symbols are analyzed and the data start value is always 0.

Conditions

Measurement views: 802.11ad Constellation, EVM vs Time, and Summary view

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AD:EVM:DATA:ANALyze:ALL:SYMBols { 0 | OFF | 1| ON }
[SENSe]:WLAN:AD:EVM:DATA:ANALyze:ALL:SYMBols?
```

Arguments

0 or OFF disables Analyze All (unchecks check box)

1 or ON enables Analyze All (checks check box)

Returns

<Boolean> 0,1

Examples

SENSe:WLAN:AD:EVM:DATA:ANALYZE:ALL:SYMBOLS? might return 0, indicating that the Data start and Data length values are considered for current analysis.

[SENSe]:WLAN:AD:EVM:DATA:LENGth

Sets or queries the length value (set in the EVM tab of the Settings control panel) in the display.

Conditions

Measurement views: 802.11ad Constellation, EVM vs Time, and Summary view

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AD:EVM:DATA:LENGth <value>
[SENSe]:WLAN:AD:EVM:DATA:LENGth?
```

Arguments

<value>=<NR1>

Returns

<NR1> = length value

Examples

SENSe:WLAN:AD:EVM:DATA:LENGTH 1000, indicating that analysis should happen for 1000 symbols in the acquired data region.

[SENSe]:WLAN:AD:EVM:DATA:STARt

Sets or queries the start value (set in the EVM tab of the Settings control panel) for the display.

Conditions

Measurement views: 802.11ad Constellation, EVM vs Time, and Summary view

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AD:EVM:DATA:STARt <value>
[SENSe]:WLAN:AD:EVM:DATA:STARt?
```

Arguments

<value>=<NR1>

Returns

<NR1> = start value

Examples

`SENSe:WLAN:AD:EVM:DATA:START 5` indicates that analysis should begin at the 5th symbol.

[SENSe]:WLAN:AD:EVM:UNITs

Sets or queries the EVM units in the EVM tab of the Settings control panel.

Conditions

Measurement views: 802.11ad Constellation, EVM vs Time, and Summary views

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AD:EVM:UNITs { DB | PERC}
[SENSe]:WLAN:AD:EVM:UNITs?
```

Arguments

`DB` will set the EVM units to decibels

`PERC` will set the EVM units to percent

Returns

See Arguments

Examples

`SENSe:WLAN:AD:EVM:UNITS PERC` sets the EVM units to percent.

[SENSe]:WLAN:AD:FILTer:ALPHa

Sets or queries the Filter Parameter for all 802.11ad measurement views, which is applicable when RRC is set as the measurement filter type (in the Modulation Params tab of the Settings control panel).

Conditions

Measurement views: All WLAN 802.11ad measurement views

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AD:FILTer:ALPHa <value>
[SENSe]:WLAN:AD:FILTer:ALPHa?
```

Arguments

`<value>=<NRf>`

Returns

`<NRf>` = filter parameter

Examples

`SENSe:WLAN:AD:FILTER:ALPHA 254.0000000000E-3` will set the filter parameter to 0.254.

[SENSe]:WLAN:AD:FILTer:MEASurement

Sets or queries the measurement filter in all the 802.11ad measurement views (set in the Modulation Params tab of the Settings control panel).

Conditions

Measurement views: All 802.11ad measurement views

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AD:FILTer:MEASurement { OFF | RRC }
[SENSe]:WLAN:AD:FILTer:MEASurement?
```

Arguments

`OFF` sets the filter to None

`RRC` sets the filter to Root raised cosine

Returns

See Arguments

Examples

`SENSe:WLAN:AD:FILTER:MEASUREMENT` might return `RRC`, indicating that the measurement filter is set to Root Raised Cosine.

[SENSe]:WLAN:AD:FREQuency:ERRor

Sets or queries the frequency error value in all the 802.11ad measurement views (set in the Analysis Params tab of the Settings control panel).

Conditions

Measurement views: All WLAN 802.11ad measurement views

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AD:FREQuency:ERRor <value>
[SENSe]:WLAN:AD:FREQuency:ERRor?
```

Arguments

`<value>=<NRf>`

Returns

`<NRf>` = frequency error in Hz

Examples

`SENSe:WLAN:AD:FREQUENCY:ERROR 6` will set the frequency error to 6.000 Hz.

[SENSe]:WLAN:AD:FREQuency:ERRor:AUTO

Enables or disables the Auto check box state for frequency error in all the 802.11ad displays in the Analysis Params tab of the Settings control panel. This setting determines whether the frequency error is set automatically (enabled) or manually (disabled).

Conditions

Measurement views: All 802.11ad measurement views

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AD:FREQuency:ERRor:AUTO { 0 | OFF | 1 | ON }
[SENSe]:WLAN:AD:FREQuency:ERRor:AUTO?
```

Arguments

`OFF` or `0` disables Frequency Error Auto (Manual mode)

`ON` or `1` enables Frequency Error Auto (Auto mode)

Returns

`<Boolean>` 0, 1

Examples

`SENSe:WLAN:AD:FREQUENCY:ERROR:AUTO OFF` disables the automatic mode for frequency error.

[SENSe]:WLAN:AD:MODUlation

Sets or queries the standard modulation type in all the 802.11ad measurement views (set in the Modulation Params tab of the Settings control panel).

Conditions

Measurement views: All 802.11ad measurement views

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AD:MODUlation {SC,CPHY}
[SENSe]:WLAN:AD:MODUlation?
```

Arguments

`SC` will set the standard to Single Carrier

`CPHY` will set the standard to control Phy

Returns

See Arguments

Examples

`SENSe:WLAN:AD:MODULATION SC` will set the standard modulation type to Single Carrier.

[SENSe]:WLAN:AD:SUMMary:CLEar:RESults (No Query Form)

Clears all scalar results in the display.

Conditions

Measurement view: 802.11ad Summary

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AD:SUMMary:CLEar:RESults
```

Arguments

None

Returns

None

Examples

`SENSe:WLAN:AD:SUMMARY:CLEAR:RESULTS` clears the result in the 802.11ad Summary display.

[SENSe]:WLAN:AD:SUMMary:LIMIt:EVM

Sets or queries EVM limit value (set in the Limit tab in the Settings control panel) for the display. This value is the threshold for pass or fail.

Conditions

Measurement view: 802.11ad Summary

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AD:SUMMary:LIMIt:EVM <Arg1,Arg2>
[SENSe]:WLAN:AD:SUMMary:LIMIt:EVM?
```

Arguments

`<Arg1>` = OFF or 0 (disables EVM limit) or ON or 1 (enables EVM limit)

`<Arg2>` = limit value (Range: 0 to –160)

Returns

`<Boolean>` 0,1 and `<NRf>` (0 to –160)

Examples

`SENSe:WLAN:AD:SUMMARY:LIMIT:EVM 1,-30` enables the EVM limit and sets the limit to –30.

[SENSe]:WLAN:AD:SUMMary:LIMIt:FREQuency:ERROr

Sets or queries the frequency error limit (set in the Limit tab of the Settings control panel) for the display. This value is the threshold for pass or fail.

Conditions

Measurement view: 802.11ad Summary

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AD:SUMMary:LIMIt:FREQuency:ERROr <Arg1,Arg2>
[SENSe]:WLAN:AD:SUMMary:LIMIt:FREQuency:ERROr?
```

Arguments

`<Arg1>` = OFF or 0 (disables Freq Error Limit) or ON or 1 (enables Freq Error Limit)

`<Arg2>` = limit value

Returns

`<Boolean>` 0,1

Examples

`SENSe:WLAN:AD:SUMMARY:LIMIT:FREQUENCY:ERROR 0,60` disables the Freq Error Limit and sets the limit to 60.

[SENSe]:WLAN:ANALysis:LENGth

Specifies the length of the analysis period to use in the WLAN analysis time settings. Analysis length is specified in either symbols or seconds, depending on the Units setting. Use `[SENSe]:WLAN:UNIT:TIME` to set the units.

Programming a specified length sets `[SENSe]:WLAN:ANALysis:AUTO` to OFF.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:ANALysis:LENGth <NRf>
[SENSe]:WLAN:ANALysis:LENGth?
```

Arguments

Floating point number that specifies the analysis length.

Range: The minimum value depends on the modulation type.

Examples

`SENSe:WLAN:ANALYSIS:LENGTH 25.625e-6` sets the analysis length to 25.625 µs.

[SENSe]:WLAN:ANALysis:LENGth:ACTual? (Query Only)

Returns the value of the actual analysis length for the WLAN analysis time settings.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:ANALysis:LENGth:ACTual?
```

Returns

Floating point number that represents the actual analysis length in seconds.

Examples

`SENSe:WLAN:ANALYSIS:LENGTH:ACTUAL?` might return `25.625E-6`, indicating that the actual analysis length is 25.625 µs.

[SENSe]:WLAN:ANALysis:LENGth:AUTO

Sets the analysis length value based on the requirements of the selected display. To set the analysis length manually, use the command.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:ANALysis:LENGth:AUTO {0|1|OFF|ON}
[SENSe]:WLAN:ANALysis:LENGth:AUTO?
```

Arguments

`ON` or `1` sets the analysis length value based on the requirements of the selected display.

`OFF` or `0` turns this function off.

Examples

`SENSe:WLAN:ANALYSIS:LENGTH:AUTO ON` sets the analysis length value automatically based on the requirements of the selected display.

[SENSe]:WLAN:ANALysis:OFFSet

Specifies or queries the analysis offset in the symbol interval in WLAN analysis parameters. This value is a percentage of the Guard Interval. The allowable range is -100% to 0%. -100% positions the FFT to start at the beginning of the Guard Interval, 0% positions it to start at the end of the Guard Interval. The default value is -50% which usually gives the best measurement results.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:ANALysis:OFFSet <NR1>
[SENSe]:WLAN:ANALysis:OFFSet?
```

Arguments

Integer that represents the analysis offset in the symbol interval, in percent terms.

Examples

`SENSe:WLAN:ANALYSIS:OFFSET -50` sets the analysis offset in the symbol interval to -50%.

[SENSe]:WLAN:AY:ANALysis:BLOCkwise:CORRection:STATe

Set or query the state of the block wise data analysis (disabled or enabled).

Conditions

Measurement views: All WLAN 802.11ay

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY:ANALysis:BLOCkwise:CORRection:STATe { OFF | ON | 0 | 1 }
[SENSe]:WLAN:AY:ANALysis:BLOCkwise:CORRection:STATe?
```

Arguments

`0` or `OFF` disables block wise correction.

`1` or `ON` enables block wise correction.

Returns

`0` or `OFF` means block wise correction is turned off.

`1` or `ON` means block wise correction is turned on.

Examples

`SENSe:WLAN:AY:ANAL:BLOC:CORR:STAT 1` turns on block wise correction.

`SENSe:WLAN:AY:ANAL:BLOC:CORR:STAT?` may return `0`, which means block wise correction is turned off.

[SENSe]:WLAN:AY:ANALysis:EQUAlizer:CONVergence

Set or query the Convergence value (set in the Advanced Parameters tab of the Settings control panel) for the 802.11ay displays.

Conditions

Measurement views: All 802.11ay

It is applicable only when Adaptive is set as the Adaptive Equalizer.

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY:ANALysis:EQUAlizer:CONVergence <value>
[SENSe]:WLAN:AY:ANALysis:EQUAlizer:CONVergence?
```

Arguments

`<value>::=<NRf>` is the Convergence value.

Returns

`<NRf>`

Examples

`SENSe:WLAN:AY:ANAL:EQUA:CONV 2.000e-3` sets the Convergence value to 2.000e-3.

`SENSe:WLAN:AY:ANAL:EQUA:CONV?` may return `2.000e-3`, indicating the Convergence value is 2.000e-3.

[SENSe]:WLAN:AY:ANALysis:DC:OFFSet:CORRection:STATe

Set or query the state of the Enable DC Offset Correction checkbox (checked or unchecked) in the Settings control panel common to all the WLAN 802.11ay displays.

Conditions

Measurement views: All WLAN 802.11ay

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY:ANALysis:DC:OFFSet:CORRection:STATe { OFF | ON | 0 | 1 }
[SENSe]:WLAN:AY:ANALysis:DC:OFFSet:CORRection:STATe?
```

Arguments

`0` or `OFF` turns off Enable DC Offset Correction.

`1` or `ON` turns on Enable DC Offset Correction.

Returns

`0` or `OFF` means that Enable DC Offset Correction is disabled.

`1` or `ON` means that Enable DC Offset Correction is enabled.

Examples

`SENSe:WLAN:AY:ANAL:DC:OFFS:CORR:STAT 0` unchecks the Enable DC Offset Correction checkbox.

`SENSe:WLAN:AY:ANAL:DC:OFFS:CORR:STAT?` may return `1`, which means that the Enable DC Offset Correction checkbox is checked

[SENSe]:WLAN:AY:ANALysis:EQUAlizer:LENGth

Set or query the Adaptive Equalizer Length (set in the Advanced Parameters tab of the Settings control panel) for the 802.11ay displays.

If you use SignalVu for scopes, this command is only available for the SignalVu version 3.10.0030 and above.)

Conditions

Measurement views: All 802.11ay

Only applies when adaptive is set as the adaptive equalizer.

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY:ANALysis:EQUAlizer:LENGth <value>
[SENSe]:WLAN:AY:ANALysis:EQUAlizer:LENGth?
```

Arguments

`<value>::=<NR1>` is the Adaptive Equalizer Length value.

Returns

`<NR1>`

Examples

`SENSe:WLAN:AY:ANAL:EQUA:LENG 25` sets Adaptive Equalizer Length to 25.

`SENSe:WLAN:AY:ANAL:EQUA:LENG?` may return `25`, which indicates that Adaptive Equalizer Length is 25.

[SENSe]:WLAN:AY:ANALysis:EQUAlizer:MODE

Set or query the Equalizer Mode (set in the Advanced Parameters tab of the Settings control panel) for 802.11ay displays.

If you use SignalVu for scopes, this command is only available for SignalVu version 3.10.0030 and above.

Conditions

Measurement views: All 802.11ay

It is applicable only when Adaptive is set as the Adaptive Equalizer.

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY:ANALysis:EQUAlizer:MODE { TRA | HOLD }
[SENSe]:WLAN:AY:ANALysis:EQUAlizer:MODE?
```

Arguments

`TRA` sets mode to Train.

`HOLD` sets mode to Hold.

Returns

`<string>` = TRA or HOLD

Examples

SENSe:WLAN:AY:ANAL:EQUA:MODE TRA sets the Equalizer Mode to Train.

SENSe:WLAN:AY:ANAL:EQUA:MODE? may return HOLD, which indicates that the Equalizer Mode is Hold.

[SENSe]:WLAN:AY:ANALysis:EQUAlizer:RESet (No Query Form)

Reset the Equalization value (button in the Advanced Parameters tab of the Settings control panel) for the 802.11ay displays.

If you use SignalVu for scopes, this command is only available for SignalVu version 3.10.0030 and above.

Conditions

Measurement views: All 802.11ay

It is applicable only when Adaptive is set as the Adaptive Equalizer.

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY:ANALysis:EQUAlizer:RESet
```

Arguments

None

Examples

SENSe:WLAN:AY:ANAL:EQUA:RES resets the Equalization value.

[SENSe]:WLAN:AY:ANALysis:EQUAlizer:TAPS

Set or query the Taps value in the Advanced Parameters tab of the Settings control panel for the 802.11ay displays.

If you use SignalVu for scopes, this command is only available for SignalVu version 3.10.0030 and above.

Conditions

Measurement views: All 802.11ay

It is applicable only when Adaptive is set as the Adaptive Equalizer.

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY:ANALysis:EQUAlizer:TAPS <value>
[SENSe]:WLAN:AY:ANALysis:EQUAlizer:TAPS?
```

Arguments

<value>::=<NR1> is the Taps value.

Returns

<NR1>

Examples

`SENSe:WLAN:AY:ANAL:EQUA:TAPS 56` sets the Taps value to 56.

`SENSe:WLAN:AY:ANAL:EQUA:TAPS?` may return `56`, which means that the Taps value is 56.

[SENSe]:WLAN:AY:ANALysis:EQUAlizer:TSRatio

Set or query the Taps/Symbols value (set in the Advanced Parameters tab of the Settings control panel) for the 802.11ay displays.

If you use SignalVu for scopes, this command is only available for SignalVu version 3.10.0030 and above.

Conditions

Measurement views: All 802.11ay

It is applicable only when Adaptive is set as the Adaptive Equalizer.

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY:ANALysis:EQUAlizer:TSRatio { ONE | TWO | FOUR }
[SENSe]:WLAN:AY:ANALysis:EQUAlizer:TSRatio?
```

Arguments

`ONE`, `TWO`, `FOUR`

Returns

`<string>` = ONE, TWO, or FOUR

Examples

`SENSe:WLAN:AY:ANAL:EQUA:TSR ONE` sets the Taps/Symbols value to 1.

`SENSe:WLAN:AY:ANAL:EQUA:TSR?` may return `FOUR`, indicating the value of Taps/Symbols is 4.

[SENSe]:WLAN:AY:ANALysis:EQUAlizer:TYPE

Set or query whether the Equalizer is set to OFF, Nonadaptive (NONA), or Adaptive (ADAPtive) in the settings control panel common to all the WLAN 802.11ay displays.

Conditions

Measurement views: All WLAN 802.11ay

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY:ANALysis:EQUAlizer:TYPE { NONE | NONA | ADAPTive }
[SENSe]:WLAN:AY:ANALysis:EQUAlizer:TYPE?
```

Arguments

`NONE` sets the equalizer to OFF.

`NONA` sets the equalizer to nonadaptive.

`ADAPtive` sets the equalizer to adaptive.

Returns

NONE means that the equalizer is set to OFF.

NONA means that the equalizer is set to Nonadaptive.

ADAPtive means that the equalizer is set to Adaptive.

Examples

SENSe:WLAN:AY:ANAL:EQUAl:TYPE ADAPtive sets the equalizer to Adaptive

SENSe:WLAN:AY:ANAL:EQUAl:TYPE? may return NONE, which means that the equalizer is set to OFF.

[SENSe]:WLAN:AY:ANALysis:LENGth

Set or query the length value of the analysis period to be used in the WLAN 802.11ay analysis time settings.

The analysis length is specified in symbols or seconds, depending on the units setting.

Conditions

Measurement views: All WLAN 802.11ay

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY:ANALysis:LENGth <NRf>
[SENSe]:WLAN:AY:ANALysis:LENGth?
```

Arguments

<NRf> is the floating point number that specifies the analysis length.

Range: the minimum value depends on the modulation type. Units: µs or s (seconds)

Returns

<NRf>

Examples

SENSe:WLAN:AY:ANAL:LENG 24.7835200000e-6 sets the analysis length 24.784 µs.

SENSe:WLAN:AY:ANAL:LENG? may return 24.7835200000e-6, which means that the analysis length is 24.784 µs.

[SENSe]:WLAN:AY:ANALysis:LENGth:ACTual? (Query Only)

Query the value of the actual analysis length for the WLAN 802.11ay analysis time settings.

Conditions

Measurement views: All WLAN 802.11ay

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY:ANALysis:LENGth:ACTual?
```

Returns

<NRf>

Units: µs or s (seconds)

Examples

SENSe:WLAN:AY:ANAL:LENG:ACT? may return 2.840e-6, indicating that the actual analysis length is 2.840 µs.

[SENSe]:WLAN:AY:ANALysis:LENGth:AUTO

Set or query the state of the analysis length (enabled or disabled) based on the requirements of the selected display.

To set the analysis length value manually, use this command.

Conditions

Measurement views: All WLAN 802.11ay

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY:ANALysis:LENGth:AUTO { OFF | ON | 0 | 1 }
[SENSe]:WLAN:AY:ANALysis:LENGth:AUTO?
```

Arguments

ON or 1 enables analysis length.

OFF or 0 disables analysis length.

Returns

ON or 1 sets analysis length to automatic (box is checked).

OFF or 0 sets analysis length to manual (box is unchecked).

Examples

SENSe:WLAN:AY:ANAL:LENG:AUTO ON sets analysis length automatically.

SENSe:WLAN:AY:ANAL:LENG:AUTO? may return OFF, which means analysis length is turned off.

[SENSe]:WLAN:AY:ANALysis:OFFSet

Set or query the value of the Analysis Offset for all the WLAN 802.11ay displays. The value is located under the Settings control panel in all WLAN 802.11ay displays.

The query returns the result in symbols or seconds, based on the unit choice selected in the analysis time tab in the WLAN 802.11ay settings control panel.

Conditions

Measurement views: All WLAN 802.11ay

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY:ANALysis:OFFSet <value>
[SENSe]:WLAN:AY:ANALysis:OFFSet?
```

Arguments

<value>::=<NRf> sets the analysis offset in the Analysis Time tab of the WLAN 802.11ay Settings control panel.

Units: µs or s (seconds)

Returns

<NRf>

Examples

SENSe:WLAN:AY:ANAL:OFFS -1.2e-6 sets the Analysis Offset to -1.200 µs.

SENSe:WLAN:AY:ANAL:OFFS? may return -1.2e-6, which means that the analysis offset is -1.200 µs.

[SENSe]:WLAN:AY:ANALysis:OFFSet:AUTO

Sets or query the state of the Auto checkbox (enabled or disabled) for the analysis offset.

This setting determines if the analysis offset displays is set automatically or manually.

Conditions

Measurement views: All WLAN 802.11ay

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY:ANALysis:OFFSet:AUTO { OFF | ON | 0 | 1 }
[SENSe]:WLAN:AY:ANALysis:OFFSet:AUTO?
```

Arguments

0 or OFF sets the analysis offset to manual (box is unchecked).

1 or ON sets the analysis offset to automatic (box is checked).

Returns

0 or OFF means that the analysis offset is manual.

1 or ON means that the analysis offset is automatic.

Examples

SENSe:WLAN:AY:ANAL:OFFS:AUTO 1 sets the analysis offset to automatic.

SENSe:WLAN:AY:ANAL:OFFS:AUTO? may return 0, which means that the analysis offset to manual.

[SENSe]:WLAN:AY:ANALysis:SFO:TRACk:STATe

Set or query the state of the SFO Tracker. When enabled, the SFO Tracker estimates and corrects Symbol Timing/ Frequency errors for each Data/GI block. When disabled, it performs a single shot correction for the entire packet, which is based on the value estimated from the STF Region.

Conditions

Measurement views: All WLAN 802.11ay

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY:ANALysis:SFO:TRACk:STATe { OFF | ON | 0 | 1 }
```

Arguments

`0` or `OFF` disables the SFO Tracker.

`1` or `ON` enables the SFO Tracker.

Returns

0 or OFF means the SFO Tracker is turned off.

1 or ON means the SFO Tracker is turned on.

Examples

`SENSe:WLAN:AY:ANAL:SFO:TRAC:STAT 0` turns off the SFO Tracker, performing a single shot correction for the entire data packet.

`SENSe:WLAN:AY:ANAL:SFO:TRAC:STAT?` may return `1`, which indicates that the SFO Tracker is turned on, correcting Symbol Timing and Frequency errors for each Data/GI block in the packet.

[SENSe]:WLAN:AY:ANALysis:SWAP:IQ

Set or query the sate of the Maximum Bursts to Average option (enabled or disabled) in the WLAN 802.11ay Summary setting control panel.

Conditions

Measurement views: WLAN 802.11ay Summary

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY:ANALysis:SWAP:IQ { OFF | ON | 0 | 1 }
[SENSe]:WLAN:AY:ANALysis:SWAP:IQ?
```

Arguments

`OFF` or `0` disables Maximum Bursts to Average.

`1` or `ON` enables Maximum Bursts to Average.

Returns

0 or OFF means Maximum Bursts to Average is turned off.

1 or ON means Maximum Bursts to Average is turned on.

Examples

`SENSe:WLAN:AY:ANAL:SWAP:IQ 0` disables the IQ Swap for the WLAN 802.11ay Summary display.

`SENSe:WLAN:AY:ANAL:SWAP:IQ?` may return `1`, which indicates that the IQ Swap is enabled for the WLAN 802.11ay Summary display.

[SENSe]:WLAN:AY:ANALysis:TIMe:UNITs

Set or query the type of time units in the WLAN 802.11ay settings.

Conditions

Measurement views: All WLAN 802.11ay

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY:ANALysis:TIMe:UNITs { SEC | SYMB }
[SENSe]:WLAN:AY:ANALysis:TIMe:UNITs?
```

Arguments

`SEC` sets time units to seconds.

`SYMB` sets time units to symbols.

Returns

`SEC` means that time units are set to seconds.

`SYMB` means that time units are set to symbols.

Examples

`SENSe:WLAN:AY:ANAL:TIM:UNIT SEC` sets time units to seconds.

`SENSe:WLAN:AY:ANAL:TIM:UNIT?` may return `SYMB`, which means that time units are set to symbols.

[SENSe]:WLAN:AY:ANALysis:TRACk:EQ:STATe

Set or query the state of the LMS Tracking Equalizer. When enabled, it performs LMS Tracking Equalization for each Data/GI block. When disabled, it provides a single shot tracking and equalization for the entire packet.

Conditions

Measurement views: All WLAN 802.11ay

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY:ANALysis:TRACk:EQ:STATe { OFF | ON | 0 | 1 }
[SENSe]:WLAN:AY:ANALysis:TRACk:EQ:STATe?
```

Arguments

`0` or `OFF` disables the LMS Tracking Equalizer.

`1` or `ON` enables the LMS Tracking Equalizer.

Returns

0 or OFF means the LMS Tracking Equalizer is turned off.

1 or ON means the LMS Tracking Equalizer is turned on.

Examples

`SENSe:WLAN:AY:ANAL:TRAC:EQ:STAT 1` turns on the LMS Tracking Equalizer for each Data/GI block.

`SENSe:WLAN:AY:ANAL:TRAC:EQ:STAT?` may return `0`, which indicates that the LMS Tracking Equalizer is turned off.

[SENSe]:WLAN:AY(:BANDwidth|:BWIDth):MODE

Set or query the mode of the Measurement BW in the Settings control panel common to all WLAN 802.11ay displays.

Conditions

Measurement views: All WLAN 802.11ay

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY(:BANDwidth|:BWIDth):MODE { MAN | AUTO | LSP}
[SENSe]:WLAN:AY(:BANDwidth|:BWIDth):MODE?
```

Arguments

Select one of the below arguments:

Argument	Description
MANual	Allows manual selection of the measurement bandwidth. A typical use is setting a narrow measurement bandwidth on a small range of frequencies to examine one signal, while rejecting the others that are present in a spectrum graph.
AUTO	Automatically selects a measurement bandwidth. The measurement chooses a bandwidth, based on other parameter settings, such as symbol rate and modulation type.
LSP	Sets the Measurement BW to Link to Span. This allows you to use the spectrum display to tune the frequency, view the signal, and then use span control to set the measurement bandwidth. This mode emulates legacy instruments.

Returns

See Arguments.

Examples

`SENSe:WLAN:AY:BAND:MODE LSP` sets the Measurement BW mode to Link to Span.

`SENSe:WLAN:AY:BAND:MODE?` may return `AUTO`, which means that the Measurement BW Mode automatically selects a measurement bandwidth.

[SENSe]:WLAN:AY(:BANDwidth|:BWIDth):TINTerval

Set or query the value of the Measurement Bandwidth (BW) present in the settings control panel common to all the WLAN 802.11ay displays.

Conditions

Measurement views: All WLAN 802.11ay

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY(:BANDwidth|:BWIDth):TINTerval <value>
[SENSe]:WLAN:AY(:BANDwidth|:BWIDth):TINTerval?
```

Arguments

`<value>::=<NRf>` specifies the measurement bandwidth.

Range: 1 to 100 Hz

Returns

`<NR3>`

Examples

`SENSe:WLAN:AY:BAND:BWID:TINT 5` sets the Measurement Bandwidth value to 5 Hz.

`SENSe:WLAN:AY:BAND:BWID:TINT?` may return `22`, which indicates that the Measurement Bandwidth value is 22 Hz.

[SENSe]:WLAN:AY:CHbw

Set or query the type of channel bandwidth present in the settings control panel common to all WLAN 802.11ay displays.

Conditions

Measurement views: All WLAN 802.11ay

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY:CHbw { BW1X | BW2X }
[SENSe]:WLAN:AY:CHbw?
```

Arguments

`BW1X` sets the channel bandwidth to 2.16 GHz.

`BW2X` sets the channel bandwidth to 4.32 GHz.

Units: GHz

Returns

BW1X means that the channel bandwidth is 2.16 GHz.

BW2X means that the channel bandwidth is 2.16 GHz.

Examples

`SENSe:WLAN:AY:CH BW1X` sets the channel bandwidth to 2.16 GHz.

`SENSe:WLAN:AY:CH?` may return `BW2X`, which indicates that the channel bandwidth is 4.32 GHz.

[SENSe]:WLAN:AY:DUPBand

Set or query the band (Upper or Lower) that is showing the results when a nonEDMG Duplicate mode is detected with 11ay SC 4.32 GHz BW settings.

Conditions

Measurement views: All WLAN 802.11ay

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY:DUPBand { LOWer | UPPer }
[SENSe]:WLAN:AY:DUPBand?
```

Arguments

`LOWer` sets the band to 2G-L for all open displays and updates the results.

`UPPer` sets the band to 2G-U for all open displays and updates the results.

Returns

`LOWer` means that the band is set to 2G-L for all open displays

`UPPer` sets the band to 2G-U for all open displays.

Examples

`SENSe:WLAN:AY:DUPB UPPer` sets the band to 2G-U for all open displays.

`SENSe:WLAN:AY:DUPB?` may return `LOWer`, which indicates that 2G-L band is set for all open displays.

[SENSe]:WLAN:AY:EVM:BURSt:COUNt

Set or query the total burst count to be analyzed.

Conditions

Measurement views: WLAN 802.11ay Summary

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY:EVM:BURSt:COUNt <value>
[SENSe]:WLAN:AY:EVM:BURSt:COUNt?
```

Arguments

`<value>::=<NR1>` is the total burst count.

Returns

`<NR1>`

Examples

`SENSe:WLAN:AY:EVM:BURSt:COUN 20` sets the total burst count to be analyzed to 20.

`SENSe:WLAN:AY:EVM:BURSt:COUN?` may return `20`, which indicates that the total burst count to be analyzed is 20.

[SENSe]:WLAN:AY:EVM:BURSt:COUNt:ENABle

Set or query the state of the Maximum Bursts to Average option (enabled or disabled) in the WLAN 802.11ay Summary setting control panel.

Conditions

Measurement views: WLAN 802.11ay Summary

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY:EVM:BURSt:COUNt:ENABle { OFF | ON | 0 | 1 }
[SENSe]:WLAN:AY:EVM:BURSt:COUNt:ENABle?
```

Arguments

`0` or `OFF` disables Maximum Bursts to Average.

`1` or `ON` enables Maximum Bursts to Average .

Returns

0 or OFF means that Maximum Bursts to Average is turned off.

1 or ON means that Maximum Bursts to Average is turned on.

Examples

`SENSe:WLAN:AY:EVM:BURS:COUN:ENAB 0` turns off Maximum Bursts to Average option.

`SENSe:WLAN:AY:EVM:BURS:COUN:ENAB?` may return `1`, which indicates that Maximum Bursts to Average option is turned on.

[SENSe]:WLAN:AY:EVM:DATA:ANALyze:ALL:SYMBols

Set or query whether the Analyze All option is enabled or disabled. When enabled, it allows all available data symbols to be analyzed.

Data Start and Data Length values are considered for current analysis. When using standard preset, it defaults to unchecked mode.

Conditions

Measurement views: WLAN 802.11ay Summary

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY:EVM:DATA:ANALyze:ALL:SYMBols { OFF | ON | 0 | 1 }
[SENSe]:WLAN:AY:EVM:DATA:ANALyze:ALL:SYMBols?
```

Arguments

`0` or `OFF` disables Analyze All.

`1` or `ON` enables Analyze All.

Returns

0 or OFF means that Analyze All is turned off.

1 or ON means that Analyze All is turned on.

Examples

`SENSe:WLAN:AY:EVM:DATA:ANAL:ALL:SYMB 1` enables data start and data length values to be considered for current analysis.

`SENSe:WLAN:AY:EVM:DATA:ANAL:ALL:SYMB?` may return `0`, indicating that data start and data length values are considered for current analysis.

[SENSe]:WLAN:AY:EVM:DATA:LENGth

Set or query whether the Analyze All checkbox is checked or unchecked. When enabled, all data symbols are analyzed and the data start value is always 0.

Conditions

Measurement views: WLAN 802.11ay Constellation, WLAN 802.11ay EVM versus Time and WLAN 802.11ay Summary displays

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY:EVM:DATA:LENGth <value>
[SENSe]:WLAN:AY:EVM:DATA:LENGth?
```

Arguments

`<value>::=<NR1>` is the number of symbols to be analyzed.

Returns

`<NR1>`

Examples

`SENSe:WLAN:AY:EVM:DATA:LENG 1000` sets the analysis to occur for 1000 symbols.

`SENSe:WLAN:AY:EVM:DATA:LENG?` may return `1000`, which means that the analysis occurs for 1000 symbols.

[SENSe]:WLAN:AY:EVM:DATA:STARt

Set or query whether the Analyze All checkbox is checked or unchecked. When enabled, all data symbols are analyzed and the data start value is always 0.

Conditions

Measurement views: WLAN 802.11ay Constellation, WLAN 802.11ay EVM versus Time and WLAN 802.11ay Summary displays

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY:EVM:DATA:STARt <value>
[SENSe]:WLAN:AY:EVM:DATA:STARt?
```

Arguments

<value>::=<NRf> is the symbol number.

Returns

<NRf>

Examples

SENSe:WLAN:AY:EVM:DATA:STAR 5 sets data analysis to begin at the 5th symbol.

SENSe:WLAN:AY:EVM:DATA:STAR? may return 5, which indicates that data analysis begins at the 5th symbol.

[SENSe]:WLAN:AY:EVM:UNITs

Set or query the type of EVM units (dB or percent).

Conditions

Measurement views: All WLAN 802.11ay

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY:EVM:UNITs { DB | PERC }
[SENSe]:WLAN:AY:EVM:UNITs?
```

Arguments

DB sets EVM units to dB.

PERC sets EVM units to %.

Returns

DB means that EVM units are in dB.

PERC means that EVM units are in %.

Examples

SENSe:WLAN:AY:EVM:UNIT PERC sets EVM units to percent (%).

SENSe:WLAN:AY:EVM:UNIT? may return DB, which indicates that EVM unit are in dB.

[SENSe]:WLAN:AY:FILTer:ALPHa

Set or query the value of the filter parameter, which is applicable when RRC is set as the measurement filter type.

Conditions

Measurement views: All WLAN 802.11ay

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY:FILTer:ALPHa <value>
[SENSe]:WLAN:AY:FILTer:ALPHa?
```

Arguments

<value>::=<NRf> is the filter parameter value.

Returns

<NRf>

Examples

SENSe:WLAN:AY:FILT:ALPH 254.0000000000e-3 sets the filter parameter value to 0.254.

SENSe:WLAN:AY:FILT:ALPH? may return 254.0000000000e-3, which means that the filter parameter value is 254.0000000000e-3.

[SENSe]:WLAN:AY:FILTer:MEASurement

Set or query type of the WLAN 802.11ay measurement filter setting.

Conditions

Measurement views: All WLAN 802.11ay

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY:FILTer:MEASurement { OFF | RRC }
[SENSe]:WLAN:AY:FILTer:MEASurement?
```

Arguments

OFF sets the measurement filter settings to None.

RRC sets the measurement filter settings to Root Raised Cosine.

Returns

OFF means that the measurement filter settings is set to None.

RRC means that the measurement filter settings is set to Root Raised Cosine.

Examples

SENSe:WLAN:AY:FILT:MEAS OFF sets the measurement filter settings to None.

SENSe:WLAN:AY:FILT:MEAS? may return RRC, which means that the measurement filter settings is Root Raised Cosine.

[SENSe]:WLAN:AY:FREQuency:ERRor

Set or query the value of the frequency error present in the Settings control panel common to all WLAN 802.11ay displays.

Conditions

Measurement views: All WLAN 802.11ay

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY:FREQuency:ERRor <value>
[SENSe]:WLAN:AY:FREQuency:ERRor?
```

Arguments

`<value>::=<NRf>` is the frequency error value.

Range: -2.5 to 2.5 Hz

Returns

`<NRf>`

Examples

`SENSe:WLAN:AY:FREQ:ERR 6` sets the frequency error value to 6.000 Hz.

`SENSe:WLAN:AY:FREQ:ERR?` may return `5`, which means that the frequency error value is 5.000 Hz.

[SENSe]:WLAN:AY:FREQuency:ERRor:AUTO

Set or query the state of the Auto checkbox (enabled or disabled) for frequency errors in the settings control panel common to all WLAN 802.11ay displays.

This setting determines if the frequency error is set automatically or manually.

Conditions

Measurement views: All WLAN 802.11ay

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY:FREQuency:ERRor:AUTO { OFF | ON | 0 | 1 }
[SENSe]:WLAN:AY:FREQuency:ERRor:AUTO?
```

Arguments

`OFF` or `0` disables the automatic frequency error mode (and sets the manual mode).

`ON` or `1` enables the automatic frequency error (and sets the auto mode).

Returns

OFF or 0 means that the automatic frequency error mode is disabled.

ON or 1 means that the automatic frequency error is enabled.

Examples

`SENSe:WLAN:AY:FREQ:ERR:AUTO OFF` disables the frequency error mode and sets the manual mode.

`SENSe:WLAN:AY:FREQ:ERR:AUTO?` may return `ON`, which means that the frequency error mode is automatic.

[SENSe]:WLAN:AY:MODUlation

Set or query the type of modulation present in the settings control panel that is common to all WLAN 802.11ay displays.

Conditions

Measurement views: All WLAN 802.11ay

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY:MODUlation AYSC
[SENSe]:WLAN:AY:MODUlation?
```

Arguments

`AYSC` sets the standard to 802.11ay Single Carrier.

Returns

AYSC

Examples

`SENSe:WLAN:AY:MOD AYSC` sets the modulation type to Single Carrier.

`SENSe:WLAN:AY:MOD?` may return `AYSC`, which means the modulation type is Single Carrier.

[SENSe]:WLAN:AY:Region

Set or query EDMG or PreEDMG region in the Settings control panel when 11ay 4.32 GHz BW is selected.

Conditions

Measurement views: All WLAN 802.11ay

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY:Region { EDMG | PRE1edmg | PRE2edmg }
[SENSe]:WLAN:AY:Region?
```

Arguments

Select one of the below arguments:

Argument	Description
EDMG	Sets the region to EDMG for all open displays.
PRE1edmg	Sets the region to PRE1edmg for all open displays.
PRE2edmg	Sets the region to PRE2edmg for all open displays.

Returns

See Arguments.

Examples

`SENSe:WLAN:AY:REG EDMG` sets the region to EDMG for all open displays when 11ay 4.32 GHz BW is selected.

SENSe:WLAN:AY:REG? may return PRE1, which indicates that the region is set to PRE1edmg for all open displays when 11ay 4.32 GHz BW is selected.

[SENSe]:WLAN:AY:SUMMary:CLEar:RESults (No Query Form)

Clears all Scalar results in the Summary display.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY:SUMMary:CLEar:RESults
```

Arguments

None

Examples

SENSe:WLAN:AY:SUMM:CLE:RES clears all Scalar results.

[SENSe]:WLAN:AY:SUMMary:LIMIt:EVM

Set or query the EVM Limit value (arg1), which is located under the limit tab in the Summary control panel, and the EVM Limit status (arg2) as disabled or enabled.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY:SUMMary:LIMIt:EVM <arg1,arg2>
[SENSe]:WLAN:AY:SUMMary:LIMIt:EVM?
```

Arguments

Number of arguments is two: <arg1,arg2>. Units: dB or %.

<arg1>:

OFF or 0 disables EVM Limit.

ON or 1 enables EVM Limit.

<arg2>:

Sets the limit EVM limit value to show pass/fail statuses for the EVM measurement result.

You must select the units (dB or %) in the EVM tab.

Returns

0 or 1, <NRf>

Examples

SENSe:WLAN802_11ad:SUMM:LIMI:EVM 1,-30, where <arg1> is 1, which indicates that Limit is enabled for the EVM Limit comparison and <arg2> is -30, which sets the EVM Limit value at -30%.

SENSe:WLAN802_11ad:SUMM:LIMI:EVM? may return 0, which indicates that Limit is disabled for EVM Limit comparison and sets the EVM Limit value at -30 dB.

[SENSe]:WLAN:AY:SUMMary:LIMIt:FREQuency:ERROr

Set or query the Frequency Error Limit value (arg1), which is located under the limit tab in the Summary control panel, and the Frequency Error Limit status (arg2) as disabled or enabled.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Sense commands

Syntax

```
[SENSe]:WLAN:AY:SUMMary:LIMIt:FREQuency:ERROr <arg1,arg2>
[SENSe]:WLAN:AY:SUMMary:LIMIt:FREQuency:ERROr?
```

Arguments

Number of arguments is two: <arg1,arg2>. Units: ppm.

<arg1>:

OFF or 0 turns off Frequency Error Limit.

ON or 1 turns on Frequency Error Limit.

<arg2>:

Specifies show pass/fail statuses for the Frequency Error Limit value.

Returns

0 or 1, <NRf>

Examples

SENSe:WLAN:AY:SUMM:LIMI:FREQ:ERRO 1,20 where <arg1> is 1, which indicates that Limit is enabled for Frequency Error Limit comparison, and <arg2> is 20, which sets the Frequency Error Limit value at 20 ppm.

SENSe:WLAN:AY:SUMM:LIMI:FREQ:ERRO? may return 1, 20, where 1 indicates that Limit is enabled for Frequency Error Limit comparison and 20 indicates that the frequency error limit value is 20 ppm.

[SENSe]:WLAN:BURSt

Specifies the burst to measure when multiple bursts are present in an acquisition, in the WLAN analysis parameters. The index number of the first burst is 1, second burst is 2, etc.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:BURSt <NR1>
[SENSe]:WLAN:BURSt?
```

Arguments

Integer that specifies which burst to measure when multiple bursts are present in an acquisition.

Examples

`SENSe:WLAN:BURST 3` sets the burst to measure in the presence of multiple bursts to the third burst.

[SENSe]:WLAN:CHANnel[:BANDwidth|:BWIDth]

Specifies or queries a nominal channel bandwidth to use for the WLAN measurements.

To automatically set the subcarrier spacing for the WLAN measurements, use the command `[SENSe]:WLAN:SCARriers:SPACing:AUTO`.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:CHANnel[:BANDwidth|:BWIDth] {BW5|BW10|BW20|BW40|BW80|BW160}
[SENSe]:WLAN:CHANnel[:BANDwidth|:BWIDth]?
```

Arguments

`BW5`, `BW10`, `BW20`, `BW40`, `BW80`, `BW160` specify a nominal channel bandwidth to use for the WLAN measurements.

Returns

Query returns one of the argument values indicating the selected nominal channel BW.

Examples

`SENSe:WLAN:CHANNEL:BANDWIDTH BW20` sets the nominal channel bandwidth to use to 20 MHz.

[SENSe]:WLAN:EQUalizer:TRAining

Specifies the method to use for estimating channel frequency response and equalization for WLAN measurments (either PREamble or PDATa). This control can be used to diagnose changes in frequency response over the signal packet.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:EQUalizer:TRAining {PREamble|PDATa}
[SENSe]:WLAN:EQUalizer:TRAining?
```

Arguments

`PREamble` The instrument uses only the Preamble to estimate channel frequency response. This response is then used to equalize the entire signal packet.

`PDATa` Preamble plus data. The instrument makes an initial channel frequency response estimate from the Preamble. Then it estimates the channel response for each data symbol using the decoded data content to derive equalization for each symbol individually. This allows compensation for time-varying channel response over the packet.

Examples

`SENSe:WLAN:EQUALIZER:TRAINING PREAMBLE` sets the method used to estimate channel frequency response and equalization to Preamble.

[SENSe]:WLAN:FFT:LENGth? (Query Only)

Returns the fixed FFT length for the WLAN measurements.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:FFT:LENGth?
```

Returns

Integer that represents the fixed FFT length for the WLAN measurements.

Examples

`SENSe:WLAN:FFT:LENGTH?` might return the fixed FFT length as 64.

[SENSe]:WLAN:FFT:SRATe? (Query Only)

Returns the FFT sample rate for the WLAN measurements, in samples per second, based on the bandwidth or subcarrier setting.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:FFT:SRATe?
```

Returns

Floating point number that represents the FFT sample rate in MS/s.

Examples

`SENSe:WLAN:FFT:SRATE?` might return `20.000e+6`, which represents an FFT sample rate of 20.000 MS/s.

[SENSe]:WLAN:FREQuency:ERRor

Sets or queries the frequency error in the WLAN analysis parameters. The entered value is used by the analysis as a fixed frequency offset. This is useful when

the exact frequency offset of the signal is known. To set the frequency error automatically, use the command `[SENSe]:WLAN:FREQuency:ERRor:AUTO`.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:FREQuency:ERRor <NRf>
[SENSe]:WLAN:FREQuency:ERRor?
```

Arguments

Floating point number that represents the fixed frequency offset.

Examples

`SENSe:WLAN:FREQUENCY:ERROR 0.02` sets the frequency error to 0.02.

[SENSe]:WLAN:FREQuency:ERRor:AUTO

Turns on the automatic determination of the frequency error in the WLAN analysis parameters. To enter a value manually when Auto is turned off, use the command `[SENSe]:WLAN:FREQuency:ERRor`.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:FREQuency:ERRor:AUTO {0|1|OFF|ON}
[SENSe]:WLAN:FREQuency:ERRor:AUTO?
```

Arguments

`OFF` or `0` sets the frequency error manually, using the `[SENSe]:WLAN:FREQuency:ERRor` command.

`ON` or `1` sets the frequency error automatically.

Examples

`SENSe:WLAN:FREQUENCY:ERROR:AUTO ON` turns on automatic determination of the frequency error.

[SENSe]:WLAN:GUARd:INTerval

Sets or queries the Guard Interval to use in the input signal in the WLAN measurements.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:GUARd:INTerval {AUTO|FOURth|EIGHth}
[SENSe]:WLAN:GUARd:INTerval?
```

Arguments

`AUTO` Auto from SIG uses the Guard Interval value extracted from the signal

`FOURth` specifies the guard interval as 1/4.

`EIGHth` specifies the guard interval as 1/8.

Examples

`SENSe:WLAN:GUARD:INTERVAL FOUR` sets the guard interval to 1/4.

[SENSe]:WLAN:MODulation:DETect:QAM1024

Sets or queries whether or not 1024QAM detection is enabled.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:MODulation:DETect:QAM1024 { 0 | 1 }
[SENSe]:WLAN:MODulation:DETect:QAM1024?
```

Arguments

`0` means detection is disabled.

`1` means detection is enabled.

Returns

See Arguments

Examples

SENSe:WLAN:MODulation:MANual DSSS1 specifies to use DSSS1 fixed modulation regardless of the signal content.

[SENSe]:WLAN:MODulation:MANual

Specifies to use a fixed modulation type, regardless of the signal content, in the WLAN analysis parameters.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:MODulation:MANual {BPSK |QPSK |QAM16 |QAM64|QAM256 |QAM1024 |DSSS1 |DSSS2 |CCK5 |
   CCK11}
[SENSe]:WLAN:MODulation:MANual?
```

Arguments

BPSK |QPSK |QAM16 |QAM64 |QAM256 |QAM1024| DSSS1 |DSSS2| CCK5 |CCK11 are fixed modulation types.

Examples

SENSe:WLAN:MODulation:MANual DSSS1 specifies to use DSSS1 fixed modulation regardless of the signal content.

[SENSe]:WLAN:MODulation:SOURce

Specifies the method of data symbol modulation identification in the WLAN analysis parameters (either Auto Detect, Manual, or Auto from SIG).

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:MODulation:SOURce [AUTO|MANual|SIG}
[SENSe]:WLAN:MODulation:SOURce?
```

Arguments

AUTO Auto Detect estimates the modulation from the data symbol IQ content.

MANual Manual allows specifying a fixed modulation type regardless of the signal content, using the command [SENSe]:WLAN:MODulation:MANual.

SIG Auto from SIG sets the modulation as indicated by the embedded SIG preamble symbol format data.

Examples

SENSe:WLAN:MODULATION:SOURCE MANUAL enables you to specify a fixed modulation type regardless of the signal content, using the command [SENSe]:WLAN:MODulation:DETect:QAM1024.

[SENSe]:WLAN:PILot:TRACking:AMPLitude:STATe

Specifies to use the pilot subcarrier to correct amplitude variations over the packet for WLAN measurements. Note: the default setting is 0 (false).

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:PILot:TRACking:AMPLitude:STATe {0|1|OFF|ON}
[SENSe]:WLAN:PILot:TRACking:AMPLitude:STATe?
```

Arguments

`ON` or `1` specifies to use the Amplitude pilot subcarrier to correct amplitude variation over the packets.

`OFF` or `0` specifies that the Amplitude pilot subcarrier is off.

Examples

`SENSe:WLAN:PILOT:TRACKING:AMPLITUDE:STATE ON` specifies to use the pilot subcarrier to correct variations.

[SENSe]:WLAN:PILot:TRACking:PHASe:STATe

Specifies to use the pilot subcarrier to correct phase variations over the packet, in WLAN measurements. Note: the default setting is 1 (true).

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:PILot:TRACking:PHASe:STATe {0|1|OFF|ON}
[SENSe]:WLAN:PILot:TRACking:PHASe:STATe?
```

Arguments

`ON` or `1` specifies to use the pilot subcarrier to correct phase variations over the packet.

`OFF` or `0` specifies that the pilot subcarrier is off.

Examples

`SENSe:WLAN:PILOT:TRACKING:PHASE:STATE ON` specifies to use the pilot subcarrier to correct phase variations over the packet.

[SENSe]:WLAN:PILot:TRACking:TIMing:STATe

Specifies to use the pilot subcarrier to correct timing variations over the packet, in WLAN measurements. Note: the default setting is 0 (false).

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:PILot:TRACking:TIMing:STATe {0|1|OFF|ON}
[SENSe]:WLAN:PILot:TRACking:TIMing:STATe?
```

Arguments

`ON` or `1` specifies to use the pilot subcarrier to correct timing variations over the packet.

`OFF` or `0` specifies that the pilot subcarrier is off.

Examples

`SENSe:WLAN:PILOT:TRACKING:TIMING:STATE ON` specifies to use the pilot subcarrier to correct timing variations over the packet.

[SENSe]:WLAN:RADix

Specifies or queries the format for displaying the symbol values in the WLAN setting preferences (either binary or HEX).

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:RADix {BINary|HEX}
[SENSe]:WLAN:RADix?
```

Arguments

`BINary` specifies to display the symbol values in binary format.

`HEX` specifies to display the symbol values in HEX format.

Examples

`SENSe:WLAN:RADIX HEX` specifies to display the symbol values in HEX format.

[SENSe]:WLAN:SCARriers

Sets or queries which subcarriers to display in the WLAN view.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:SCARriers {ALL|PILots|DATA|RANGEe|SINGle}
[SENSe]:WLAN:SCARriers?
```

Arguments

`ALL` specifies that all subcarriers will be displayed.

`PILot` specifies that only the pilot subcarriers will be displayed.

`DATA` specifies that only the data subcarriers will be displayed.

`RANGe` specifies that the subcarriers within a specified range will be displayed.

`SINGle` specifies that only a single subcarrier will be displayed.

Examples

`SENSe:WLAN:SCARRIERS ALL` sets the WLAN view to display all subcarriers.

[SENSe]:WLAN:SCARriers:RANGe:STARt

Sets or queries the start value of the subcarrier range to be displayed in the WLAN view.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:SCARriers:RANGe:STARt <NR1>
[SENSe]:WLAN:SCARriers:RANGe:STARt?
```

Arguments

Integer

Examples

`SENSe:WLAN:SCARRIERS:RANGE:START 20` sets the subcarrier start range to 20.

[SENSe]:WLAN:SCARriers:RANGe:STOP

Sets or queries the stop value of the subcarrier range to be displayed in the WLAN view.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:SCARriers:RANGe:STOP <NR1>
[SENSe]:WLAN:SCARriers:RANGe:STOP?
```

Arguments

Integer

Examples

`SENSe:WLAN:SCARRIERS:RANGE:STOP 25` sets the subcarrier stop range to 25.

[SENSe]:WLAN:SCARriers:SINGle:INDex

Specifies or queries the single subcarrier to display in the WLAN view.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:SCARriers:SINGle:INDex <NR1>
[SENSe]:WLAN:SCARriers:SINGle:INDex?
```

Arguments

Integer

Examples

`SENSe:WLAN:SCARRIERS:SINGLE:INDEX -10` sets the single subcarrier for display to -10.

[SENSe]:WLAN:SCARriers:SPACing

Sets or queries the subcarrier spacing in the WLAN measurements.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:SCARriers:SPACing <NRf>
[SENSe]:WLAN:SCARriers:SPACing?
```

Arguments

NR3 is a floating point number that specifies the subcarrier spacing.

Returns

SENSe:WLAN:SCARRIERs:SPACING 312.5E+3 sets the subcarrier spacing to 312.5 kHz.

[SENSe]:WLAN:SCARriers:SPACing:AUTO

When turned on, automatically sets the subcarrier spacing in the WLAN measurements.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:SCARriers:SPACing:AUTO {0|1|OFF|ON}
[SENSe]:WLAN:SCARriers:SPACing:AUTO?
```

Arguments

OFF or 0 turns the automatic subcarrier spacing off.

ON or 1 turns the automatic subcarrier spacing on.

Examples

SENSe:WLAN:SCARriers:SPACing:AUTO ON turns the automatic subcarrier spacing on.

[SENSe]:WLAN:SIGNal:STANdard:TYPE

Sets or queries the WLAN signal standard.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Additionally, the N argument requires Option 24, and the AC argument requires Option 25.

Group

Sense commands

Syntax

```
[SENSe]:WLAN:SIGNal:STANdard:TYPE {A|B|G|J|P|N|AC}
[SENSe]:WLAN:SIGNal:STANdard:TYPE?
```

Arguments

A sets the WLAN signal standard to 802.11a.

B sets the WLAN signal standard to 802.11b.

G sets the WLAN signal standard to 802.11g.

J sets the WLAN signal standard to 802.11j.

P sets the WLAN signal standard to 802.11p.

N sets the WLAN signal standard to 802.11n. Requires Option 24 in addition to Option 23.

AC sets the WLAN signal standard to 802.11ac. Requires Option 25 in addition to Option 23.

Examples

SENSe:WLAN:SIGNAL:STANdARD:TYPE G sets the WLAN signal standard to 802.11g.

[SENSe]:WLAN:SubCARrier:DERotation

Turning on this state allows some displays to show subcarriers with or without the Gamma subcarrier phase rotation removed. Gamma phase rotation is applied to 802.11n and 802.11ac subcarriers in defined subranges depending on the Channel Bandwidth selected (only for Channel Bandwidths of 40 MHz and above).

When derotation is turned off, the rotation is not removed, which provides a direct view of the physical modulation on the channel. With derotation turned on, the rotation is removed, allowing easier decoding of the underlying data content.

Only the Constellation and Symbol Table results are affected by this control.

NOTE: This control is only shown for 802.11n and 802.11ac standard selections for channel bandwidths greater than or equal to 40 MHz. It is not present for all others.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:SubCARrier:DERotation {0|1|OFF|ON}
[SENSe]:WLAN:SubCARrier:DERotation?
```

Arguments

OFF or 0 turns the subcarrier derotation off.

ON or 1 turns the subcarrier derotation on.

Examples

SENSe:WLAN:SUBCARRIER:DEROTATION ON turns the subcarrier derotation on.

[SENSe]:WLAN:SUMMary:CLEar:RESults (No Query Form)

Clears the results in the WLAN Summary measurement.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:SUMMary:CLEar:RESults
```

Arguments

None

Examples

`SENSe:WLAN:SUMMARY:CLEAR:RESULTS` clears the results in the measurement.

[SENSe]:WLAN:SUMMary:EVM:RMS:AVERage:COUNt

Specifies the maximum number of bursts to average for the WLAN Summary EVM display. The burst average function must be enabled using the command `[SENSe]:WLAN:SUMMary:EVM:RMS:AVERage:COUNt:ENABle`.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:SUMMary:EVM:RMS:AVERage:COUNt <NR1>
[SENSe]:WLAN:SUMMary:EVM:RMS:AVERage:COUNt?
```

Arguments

Integer

Examples

`SENSe:WLAN:SUMMary:EVM:RMS:AVERage:COUNt 12` sets the maximum number of bursts to average to 12.

[SENSe]:WLAN:SUMMary:EVM:RMS:AVERage:COUNt:ENABle

Enables the burst average function in the WLAN Summary EVM settings. To specify the maximum number of bursts to average, use the command `[SENSe]:WLAN:SUMMary:EVM:RMS:AVERage:COUNt`.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:SUMMary:EVM:RMS:AVERage:COUNt:ENABle {0|1|OFF|ON}
[SENSe]:WLAN:SUMMary:EVM:RMS:AVERage:COUNt:ENABle?
```

Arguments

`ON` or `1` turns the burst average function on.

`OFF` or `0` turns the burst average function off.

Examples

`SENSe:WLAN:SUMMARY:EVM:RMS:AVERAGE:COUNT:ENABLE ON` turns the burst average function on.

[SENSe]:WLAN:SWAP:IQ

Specifies to swap the I and Q components of a signal in the WLAN analysis parameters. Doing so compensates the input signal for spectral inversion.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:SWAP:IQ {0|1|OFF|ON}
[SENSe]:WLAN:SWAP:IQ?
```

Arguments

`OFF` or `0` turns the Swap I and Q function off.

`ON` or `1` turns the Swap I and Q function on.

Examples

`SENSe:WLAN:SWAP:IQ ON` swaps the I and Q components of the signal.

[SENSe]:WLAN:SYMBols|CHIPs

Specifies or queries which symbols or chips are displayed in the WLAN Symbol Table.

Use `CHIPs` for 802.11b and `SYMBols` for all other standards. The DAT, PRAM and HEAD enumerations apply to 802.11b only.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:SYMBols|CHIPs {ALL|RANGe|SINGle|DAT|PRAM|HEAD}
[SENSe]:WLAN:SYMBols|CHIPs?
```

Arguments

`ALL` specifies that all symbols or chips will be displayed.

`RANGe` specifies that a range of symbols or chips will be displayed.

`SINGle` specifies that single symbols or chips will be displayed.

`DAT` specifies that the Data symbols or chips will be displayed. Applies to 802.11b only.

PRAM specifies that Preamble symbols or chips will be displayed. Applies to 802.11b only.

HEAD specifies that Header symbols or chips will be displayed. Applies to 802.11b only.

Examples

SENSe:WLAN:SYMBOLS ALL specifies that all symbols or chips will be displayed in the WLAN Symbol Table.

[SENSe]:WLAN:SYMBols|CHIPs:MAX

Sets or queries the value for the maximum number of symbols/chips to analyze for the WLAN Symbol Table.

Use CHIPs for 802.11b and SYMBols for all other standards.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:SYMBols|CHIPs:MAX <NR1>
[SENSe]:WLAN:SYMBols|CHIPs:MAX?
```

Arguments

Integer

Examples

SENSe:WLAN:SYMBOLS:MAX 50 specifies a maximum value of 50 symbols to analyze for the WLAN Symbol Table.

[SENSe]:WLAN:SYMBols|CHIPs:MAX:STATe

Turns on or off the “maximum data symbols to analyze” feature for the WLAN Symbol Table. To specify the maximum number of symbols/chips to analyze, use the command.

Use CHIPs for 802.11b and SYMBols for all other standards.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:SYMBols|CHIPs:MAX:STATe {0|1|OFF|ON}
[SENSe]:WLAN:SYMBols|CHIPs:MAX:STATe?
```

Arguments

ON or 1 turns on the “maximum data symbols to analyze” feature.

OFF or 0 disables the “maximum data symbols to analyze” feature.

Examples

SENSe:WLAN:SYMBOLS:MAX:STATE ON turns on “maximum data symbols to analyze” feature for the WLAN Symbol Table.

[SENSe]:WLAN:SYMBols|CHIPs:RANGe:COUNt

Sets or queries the number of symbols or chips to display when displaying a range results in the WLAN Symbol Table.

Use CHIPs for 802.11b and SYMBols for all other standards.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:SYMBols|CHIPs:RANGe:COUNt <NR1>
[SENSe]:WLAN:SYMBols|CHIPs:RANGe:COUNt?
```

Arguments

Integer that represents the number of symbols/chips to displayl

Examples

SENSe:WLAN:SYMBOLS:RANGE:COUNT 4 sets the number of symbols or chips to display to 4 in the WLAN Symbol Table.

[SENSe]:WLAN:SYMBols|CHIPs:RANGe:STARt

Sets or queries the symbols/chips start value when displaying a range of results in the WLAN Symbol Table.

Use CHIPs for 802.11b and SYMBols for all other standards.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:SYMBols|CHIPs:RANGe:STARt <NR1>
[SENSe]:WLAN:SYMBols|CHIPs:RANGe:STARt?
```

Arguments

Integer

Examples

SENSe:WLAN:SYMBOLS:RANGE:START 40 sets the start value of the range to 40.

[SENSe]:WLAN:SYMBols|CHIPs:RANGe:STOP

Sets or queries the symbols or chips stop value when displaying a range of results in the WLAN Symbol Table.

Use `CHIPs` for 802.11b and `SYMBols` for all other standards.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:SYMBols|CHIPs:RANGe:STOP <NR1>
[SENSe]:WLAN:SYMBols|CHIPs:RANGe:STOP?
```

Arguments

Integer

Examples

`SENSe:WLAN:SYMBOLS:RANGE:STOP 60` sets the stop value of the range to 60.

[SENSe]:WLAN:SYMBols|CHIPs:SINGle:INDex

Sets or queries the symbol or chip to use when displaying results from a single symbol in the WLAN Symbol Table.

Use `CHIPs` for 802.11b and `SYMBols` for all other standards.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:SYMBols|CHIPs:SINGle:INDex <NR1>
[SENSe]:WLAN:SYMBols|CHIPs:SINGle:INDex?
```

Arguments

Integer

Examples

`SENSe:WLAN:CHIPs:SINGle:INDex 26` specifies to display symbol 26.

[SENSe]:WLAN:UNIT:FREQuency

Specifies or queries whether the displayed frequency units are frequency (Hz) or subcarrier in the WLAN settings.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:UNIT:FREQuency {FREQuency|SCARrier}
[SENSe]:WLAN:UNIT:FREQuency?
```

Arguments

`FREQuency` specifies that the displayed frequency units will be Hz.

`SCARrier` specifies that the displayed frequency units will be subcarrier.

Examples

`SENSe:WLAN:UNIT:FREQUENCY FREQ` specifies that the frequency units in the display will be Hz.

[SENSe]:WLAN:UNIT:TIME

Specifies or queries whether the displayed time units are seconds, symbols or chips in the WLAN settings. Chips is only a valid option for 802.11b, and Symbols is valid for all other standards.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:UNIT:TIME {SEConds|SYMBols|CHIPs}
[SENSe]:WLAN:UNIT:TIME?
```

Arguments

`SEConds` specifies that the displayed time units are to be seconds

`SYMBols` specifies that the displayed time units are to be symbols

`CHIPs` specifies that the displayed time units are to be chips (valid only for 802.11b)

Examples

`SENSe:WLAN:UNIT:TIME SYMBOLS` specifies that the displayed time units are to be symbols.

SOURce:TXGain:POWer

Sets or queries the output power level of the tracking generator.

Conditions

Measurement view: Transmission Gain

Group

Source commands

Syntax

```
SOURce:TXGain:POWer <value>
```

Arguments

`<value>::=<NRf>` is the tracking generator output power.

Range: –3 dBm to –43 dBm.

Returns

See Arguments.

Examples

`SOURCE:TXGAIN:POWER -10` sets the tracking generator output power to –10 dBm.

*SRE

Sets or queries the value of the Service Request Enable Register (SRER).

Refer to Status and Events (on page 2740) for the register information.

Conditions

Measurement views: All

Group

IEEE common commands

Syntax

```
*SRE <value>
*SRE?
```

Related Commands

*CLS (on page 664)

*ESE (on page 1059)

*ESR? (on page 1059)

*STB? (on page 2531)

Arguments

`<value>::<NR1>` is a value in the range from 0 to 255.

The binary bits of the SRER are set according to this value.

Using an out-of-range value causes an execution error.

Examples

`*SRE48` sets binary 00110000 in the bits of the SRER.

`*SRE?` might return 32, indicating that binary value 00100000 has been set in the bits of the SRER.

Status commands

STATus:ACPower:EVENts? (Query Only)

Returns the current events and status conditions for the Channel power and ACPR measurement.

Conditions

Measurement views: Channel power and ACPR

Group

Status commands

Syntax

`STATus:ACPower:EVENts?`

Arguments

None

Returns

`<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where

`<ecode>::<NR1>` is the error/event code (-32768 to 32767).

`<edesc>::<string>` is the description on the error/event.

`<einfo>::<string>` is the additional information on the error/event. If there is no error, the response is 0, "No events to report".

Examples

`STATUS:ACPOWER:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:{AM|FM|PM}:EVENts? (Query Only)

Returns the current events and status conditions for the AM/FM/PM measurement.

Conditions

Measurement views: AM, FM, PM

Group

Status commands

Syntax

`STATus:{AM|FM|PM}:EVENts?`

Arguments

None

Returns

`<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where

<ecode>::<NR1> is the error/event code (-32768 to 32767).

<edesc>::<string> is the description on the error/event.

<einfo>::<string> is the additional information on the error/event. If there is no error, the response is 0, "No events to report".

Examples

STATUS:AM:EVENTS? might return 12026,"Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually in the AM measurement.

STATus:AUDio:SPECtrum:EVENts? (Query Only)

Returns the current events and status conditions for the audio measurement.

Conditions

Measurement views: Audio Spectrum

Group

Status commands

Syntax

STATus:AUDio:SPECtrum:EVENts?

Arguments

None

Returns

<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

Where:

<ecode>::<NR1> is the error/event code (-32768 to 32767).

<edesc>::<string> is the description on the error/event.

<einfo>::<string> is the additional information on the error/event. If there is no error, the response is 0, "No events to report".

Examples

STATUS:AUDIO:SPECTRUM:EVENTS? might return 12026,"Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually.

STATus:AUDio:SUMMary:EVENts? (Query Only)

Returns the current events and status conditions for the audio measurement.

Conditions

Measurement views: Audio Summary

Group

Status commands

Syntax

```
STATus:AUDio:SUMMary:EVENts?
```

Arguments

None

Returns

```
<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}
```

Where:

`<ecode>::<NR1>` is the error/event code (-32768 to 32767).

`<edesc>::<string>` is the description on the error/event.

`<einfo>::<string>` is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples

`STATUS:AUDIO:SUMMARY:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:AVTime[:MEASview<y>]:EVENts? (Query Only)

Returns the current events and status conditions for the Amplitude versus Time measurement.

Conditions

Measurement views: Amplitude versus Time

Group

Status commands

Syntax

```
STATus:AVTime[:MEASview<y>]:EVENts?
```

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

```
<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}
```

Where

`<ecode>::<NR1>` is the error/event code (-32768 to 32767).

`<edesc>::<string>` is the description on the error/event.

`<einfo>::<string>` is the additional information on the error/event. If there is no error, the response is 0, "No events to report".

Examples

`STATUS:AVTIME:MEASVIEW1:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually for Amplitude versus Time Measurement View 1.

STATus:BIBEmissions:EVENts? (Query Only)

Returns the current events and status conditions for the Bluetooth InBand Emission measurement.

Conditions

Measurement views: Bluetooth InBand Emission

Group

Status commands

Syntax

`STATus:BIBEmissions:EVENts?`

Arguments

None

Returns

`<ecode>, "<edesc>[,<einfo>]"{,"<edesc>[,<einfo>]"}`

Where:

`<ecode> ::= <NR1>` is the error/event code (-32768 to 32767).

`<edesc> ::= <string>` is the description on the error/event.

`<einfo> ::= <string>` is the additional information on the error/event. If there is no error, the response is 0, "No events to report".

Examples

`STATUS:BIBEMISSIONS:EVENTS?` might return `0` indicating no events to report.

STATus:BLUEtooth:CONSte:EVENts? (Query Only)

Returns the current events and status conditions for the Bluetooth Constellation measurement.

Conditions

Measurement views: Bluetooth Constellation

Group

Status commands

Syntax

`STATus:BLUEtooth:CONSte:EVENts?`

Arguments

None

Returns

`<ecode>, "<edesc>[<einfo>]"{, <ecode>,"<edesc>[<einfo>]"}`

Where

`<ecode> ::= <NR1>` is the error/event code (-32768 to 32767).

`<edesc> ::= <string>` is the description of the error/event.

<einfo> ::= <string> is the additional information of the error/event.

Examples

STATus:BLUEtooth:CONSte:EVENts? might return 12026,"Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually.

STATus:BLUEtooth:EDIAgram:EVENts? (Query Only)

Returns the current events and status conditions for the eye diagram measurement.

Conditions

Measurement views: Bluetooth Eye Diagram

Group

Status commands

Syntax

STATus:BLUEtooth:EDIAgram:EVENts?

Arguments

None

Returns

<ecode>, "<edesc>[<einfo>]"{, <ecode>,"<edesc>[<einfo>]"}

Where

<ecode> ::= <NR1> is the error/event code (-32768 to 32767).

<edesc> ::= <string> is the description of the error/event.

<einfo> ::= <string> is the additional information of the error/event. If there is no error, the response is 0, "No events to report".

Examples

STATus:BLUEtooth:EDIAgram:EVENts? might return 12026,"Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually.

STATus:BLUEtooth:FDVTime:EVENts? (Query Only)

Returns the current events and status conditions for the Frequency Deviation versus Time measurement.

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Status commands

Syntax

STATus:BLUEtooth:FDVTime:EVENts?

Arguments

None

Returns

<ecode>, "<edesc>[,<einfo>]"{,"<edesc>[,<einfo>]"}

Where:

<ecode> ::= <NR1> is the error/event code (-32768 to 32767).

<edesc> ::= <string> is the description on the error/event.

<einfo> ::= <string> is the additional information on the error/event. If there is no error, the response is 0, "No events to report".

Examples

STATus:BLUEtooth:FDVTime:EVENts? might return 12026,"Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually.

STATus:BLUEtooth:SUMMary:EVENts (Query Only)

Returns the current events and Status conditions for the Bluetooth Summary measurements.

Conditions

Measurement view: Bluetooth Summary display

Group

Status commands

Syntax

STATus:BLUEtooth:SUMMary:EVENts?

Arguments

None

Returns

<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"} where

<ecode>:: <NR1> is the error/event code (-32768 to 32767).

<edesc>:: <String> is the description of the error/event.

<einfo>:: <String> is the additional information on the error/event If there is no error, then response is 0, “No events to report.”

Examples

STATUS:BLUETOOTH:SUMMARY:EVENTS? might return 12026, "Acq Sampling params: manual control".

STATus:CCDF:EVENts? (Query Only)

Returns the current events and status conditions for the CCDF measurement.

Conditions

Measurement views: CCDF

Group

Status commands

Syntax

```
STATus:CCDF:EVENts?
```

Arguments

None

Returns

```
<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}
```

Where

`<ecode>::<NR1>` is the error/event code (-32768 to 32767).

`<edesc>::<string>` is the description on the error/event.

`<einfo>::<string>` is the additional information on the error/event. If there is no error, the response is 0, "No events to report".

Examples

`STATUS:CCDF:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:CONSte[:MEASview<y>]:EVENts? (Query only)

Returns the current events and status conditions for the constellation measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Constellation

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Constellation).

Group

Status commands

Syntax

```
STATus:CONSte[:MEASview<y>]:EVENts?
```

Arguments

None

Returns

```
<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}
```

Where

`<ecode>::<NR1>` is the error or event code (-32768 to 32767).

`<edesc>::<string>` is the description on the error/event.

`<einfo>::<string>` is the additional information on the error or event. If there is no error, the response is 0, "No events to report".

Examples

`STATus:CONSTe:MEASview2:EVENts?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually for measview 2 of constellation.

STATus:DIQVtime[:MEASview<y>]:EVENts? (Query only)

Returns the current events and status conditions for the Demod I&Q versus Time measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Demod I&Q versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod I&Q versus Time).

Group

Status commands

Syntax

`STATus:DIQVtime[:MEASview<y>]:EVENts?`

Arguments

None

Returns

`<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where

`<ecode>::<NR1>` is the error or event code (-32768 to 32767).

`<edesc>::<string>` is the description on the error or event.

`<einfo>::<string>` is the additional information on the error or event. If there is no error, the response is 0, "No events to report".

Examples

`STATus:DIQVtime:MEASview1:EVENts?` might return `12026, "Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually for measurement view 1.

STATus:DPX:EVENts? (Query Only)

Returns the current events and status conditions for the DPX, Spectrum measurement.

Conditions

Measurement views: DPX, Spectrum

Group

Status commands

Syntax

`STATus:DPX:EVENts?`

Arguments

None

Returns

<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

Where

<ecode>::<NR1> is the error/event code (-32768 to 32767).

<edesc>::<string> is the description on the error/event.

<einfo>::<string> is the additional information on the error/event. If there is no error, the response is 0, "No events to report".

Examples

STATUS:DPX:EVENTS? might return 12026,"Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually.

STATus:EDIagram[:MEASview<y>]:EVENts? (Query only)

Returns the current events and status conditions for the eye diagram measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Eye diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Eye diagram).

Group

Status commands

Syntax

STATus:EDIagram[:MEASview<y>]:EVENts?

Arguments

None

Returns

<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

Where

<ecode>::<NR1> is the error or event code (-32768 to 32767).

<edesc>::<string> is the description on the error or event.

<einfo>::<string> is the additional information on the error or event.

If there is no error, the response is 0, "No events to report".

Examples

STATus:EDIagram:MEASview2:EVENts? might return 12026, "Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually.

STATus:EMC:EVENts? (Query Only)

Returns the current events and status conditions for the EMC display.

Conditions

Measurement view: EMC - EMI Display

Group

Status commands

Syntax

`STATus:EMC:EVENts?`

Arguments

None

Returns

`<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where

`<ecode>::<NR1>` is the error/event code (-32768 to 32767).

`<edesc>::<string>` is the description on the error/event.

`<einfo>::<string>` is the additional information on the error/event. If there is no error, the response is 0, "No events to report".

Examples

`STATUS:EMC:EVENTS?` might return `11037,"Ext corrections enabled for traces."`

STATus:EVM[:MEASview<y>]:EVENts? (Query only)

Returns the current events and status conditions for the EVM versus Time measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: EVM versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod EVM versus Time).

Group

Status commands

Syntax

`STATus:EVM[:MEASview<y>]:EVENts?`

Arguments

None

Returns

<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

Where

`<ecode>::<NR1>` is the error or event code (-32768 to 32767).

<edesc>::<string> is the description on the error or event.

<einfo>::<string> is the additional information on the error or event. If there is no error, the response is 0, "No events to report".

Examples

STATus:EVM:MEASview2:EVENts? might return 12026, "Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually.

STATus:FDVTime[:MEASview<y>]:EVENts? (Query only)

Returns the current events and status conditions for the Frequency deviation versus Time measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Frequency deviation versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Frequency deviation versus Time).

Group

Status commands

Syntax

STATus:FDVTime:MEASview<y>:EVENts?

Arguments

None

Returns

<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

Where

<ecode>::<NR1> is the error or event code (-32768 to 32767).

<edesc>::<string> is the description on the error or event.

<einfo>::<string> is the additional information on the error or event. If there is no error, the response is 0, "No events to report".

Examples

STATUS:FDVTIME:MEASview2:EVENTS? might return 12026, "Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually.

STATus:{FSETtling|PSETtling}:EVENts? (Query Only)

Returns the current events and status conditions for the specified settling time measurement.

Conditions

Measurement views: Frequency and Phase Settling Time

Group

Status commands

Syntax

```
STATus:{FSETtling|PSETtling}:EVENts?
```

Arguments

None

Returns

```
<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}
```

Where

`<ecode>::<NR1>` is the error/event code (-32768 to 32767).

`<edesc>::<string>` is the description on the error/event.

`<einfo>::<string>` is the additional information on the error/event. If there is no error, the response is 0, "No events to report".

Examples

`STATUS:{FSETTLING|PSETTLING}:EVENTS?` might return `2048,"Done saving - restart with Stop, then Run"`, indicating that the acquisition has been completed.

STATus:FVTime[:MEASview<y>]:EVENts? (Query Only)

Returns the current events and status conditions for the Frequency versus Time measurement.

Conditions

Measurement views: Frequency versus Time

Group

Status commands

Syntax

```
STATus:FVTime[:MEASview<y>]:EVENts?
```

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

```
<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}
```

Where

`<ecode>::<NR1>` is the error/event code (-32768 to 32767).

`<edesc>::<string>` is the description on the error/event.

`<einfo>::<string>` is the additional information on the error/event. If there is no error, the response is 0, "No events to report".

Examples

`STATUS:FVTIME:MEASview1:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:IQVTime[:MEASview<y>]:EVENts? (Query Only)

Returns the current events and status conditions for the RF I&Q versus Time measurement.

Conditions

Measurement views: RF I&Q versus Time

Group

Status commands

Syntax

`STATus:IQVTime[:MEASview<y>]:EVENts?`

Arguments

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

`<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where

`<ecode>::<NR1>` is the error/event code (-32768 to 32767).

`<edesc>::<string>` is the description on the error/event.

`<einfo>::<string>` is the additional information on the error/event. If there is no error, the response is 0, "No events to report".

Examples

`STATUS:IQVTIME:MEASview1:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:LTE:ACLR:EVENts? (Query Only)

Returns the current events and status conditions for the LTE ACLR measurement.

Conditions

Measurement view: LTE ACLR

Group

Status commands

Syntax

`STATus:LTE:ACLR:EVENts?`

Returns

`<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where

`<ecode>::<NR1>` is the error/event code (-32768 to 32767).

`<edesc>::<string>` is the description of the error/event.

<einfo>::<string> is the additional information on the error/event. If there is no error, then response is 0, “No events to report.”

Examples

STATUS:LTE:ACLR:EVENTS? might return 12006, "Not aligned",2008,"Not calibrated",12007,"unaligned data",12009,"Data from uncalibrated instrument".

STATus:LTE:CHSPectrum:EVENts? (Query Only)

Returns the current events and status conditions for the LTE Channel Spectrum measurement.

Conditions

Measurement view: LTE Channel Spectrum

Group

Status commands

Syntax

STATus:LTE:CHSPectrum:EVENts?

Returns

<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

Where

<ecode>::<NR1> is the error/event code (-32768 to 32767).

<edesc>::<string> is the description of the error/event.

<einfo>::<string> is the additional information on the error/event. If there is no error, then response is 0, “No events to report.”

Examples

STATUS:LTE:CHSPECTRUM:EVENTS? might return 12006,"Not aligned",2008,"Not calibrated".

STATus:LTE:CONSte:EVENts? (Query Only)

Returns the current events and status condition for the LTE Constellation measurement.

Conditions

Measurement view: LTE Constellation

Group

Status commands

Syntax

STATus:LTE:CONSte:EVENts?

Returns

<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

Where

<ecode>::<NR1> is the error/event code (-32768 to 32767).

<edesc>::<string> is the description of the error/event.

<einfo>::<string> is the additional information on the error/event. If there is no error, then response is 0, “No events to report.”

Examples

STATUS:LTE:CONSTE:EVENTS? might return 12006,"Not aligned",2008,"Not calibrated",11090,"Recovery done on PSS/SSS on the center 62, "106 - AnaSynchronization Sequence not found".

STATus:LTE:PVTime:EVENts? (Query Only)

Returns the current events and status conditions for the LTE Power vs Time measurement.

Conditions

Measurement view: LTE Power vs Time

Group

Status commands

Syntax

STATus:LTE:PVTime:EVENts?

Returns

<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

Where

<ecode>::<NR1> is the error/event code (-32768 to 32767).

<edesc>::<string> is the description of the error/event.

<einfo>::<string> is the additional information on the error/event. If there is no error, then response is 0, “No events to report.”

Examples

STATUS:LTE:PVTIME:EVENTS? might return 12006,"Not aligned",2008,"Not calibrated".

STATus:MCPower:EVENts? (Query Only)

Returns the current events and status conditions for the MCPR (multi-carrier ACPR) measurement.

Conditions

Measurement views: MCPR

Group

Status commands

Syntax

STATus:MCPower:EVENts?

Arguments

None

Returns

<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

Where

<ecode>::<NR1> is the error/event code (-32768 to 32767).

<edesc>::<string> is the description on the error/event.

<einfo>::<string> is the additional information on the error/event. If there is no error, the response is 0, "No events to report".

Examples

STATUS:MCPOWER:EVENTS? might return 12026,"Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually.

STATus:MERRor[:MEASview<y>]:EVENts? (Query only)

Returns the current events and status conditions for the Magnitude error versus Time measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Magnitude error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Magnitude error versus Time).

Group

Status commands

Syntax

STATus:MERRor[:MEASview<y>]:EVENts?

Arguments

None

Returns

<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

Where

<ecode>::<NR1**>** is the error or event code (-32768 to 32767).

<edesc>::<string> is the description on the error or event.

<einfo>::<string> is the additional information on the error or event. If there is no error, the response is 0, "No events to report".

Examples

STATUS:MERRor:MEASview2:EVENTS? might return 12026, "Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually.

STATus:OBWidth:EVENts? (Query Only)

Returns the current events and status conditions for the Occupied Bandwidth (OBW) measurement.

Conditions

Measurement views: Occupied Bandwidth

Group

Status commands

Syntax

`STATus:OBWidth:EVENts?`

Arguments

None

Returns

`<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where

`<ecode>::<NR1>` is the error/event code (-32768 to 32767).

`<edesc>::<string>` is the description on the error/event.

`<einfo>::<string>` is the additional information on the error/event. If there is no error, the response is 0, "No events to report".

Examples

`STATUS:OBWIDTH:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:OFDM:CONSte:EVENts? (Query Only)

Returns the current events and status condition for the OFDM Constellation measurement.

Conditions

Measurement views: OFDM

Group

Status commands

Syntax

`STATus:OFDM:CONSte:EVENts?`

Arguments

None

Returns

`<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where

`<ecode>::<NR1>` is the error/event code (-32768 to 32767).

`<edesc>::<string>` is the description on the error/event.

`<einfo>::<string>` is the additional information on the error/event. If there is no error, the response is 0, "No events to report".

Examples

`STATUS:OFDM:CONSTE:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:OFDM:EVM:EVENts? (Query Only)

Returns the current events and status condition for the OFDM EVM (Error Vector Magnitude) versus Time measurement.

Conditions

Measurement views: OFDM

Group

Status commands

Syntax

`STATus:OFDM:EVM:EVENts?`

Arguments

None

Returns

`<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where

`<ecode>::<NR1>` is the error/event code (-32768 to 32767).

`<edesc>::<string>` is the description on the error/event.

`<einfo>::<string>` is the additional information on the error/event. If there is no error, the response is 0, "No events to report".

Examples

`STATUS:OFDM:EVM:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:OFDM:MERRor:EVENts? (Query Only)

Returns the current events and status condition for the OFDM Magnitude versus Time measurement.

Conditions

Measurement views: OFDM

Group

Status commands

Syntax

`STATus:OFDM:MERRor:EVENts?`

Arguments

None

Returns

<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

Where

<ecode>::<NR1> is the error/event code (-32768 to 32767).

<edesc>::<string> is the description on the error/event.

<einfo>::<string> is the additional information on the error/event. If there is no error, the response is 0, "No events to report".

Examples

STATUS:OFDM:MERROR:EVENTS? might return 12026,"Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually.

STATus:OFDM:PERRor:EVENts? (Query Only)

Returns the current events and status condition for the OFDM Phase error versus Time measurement.

Conditions

Measurement views: OFDM

Group

Status commands

Syntax

STATus:OFDM:PERRor:EVENts?

Arguments

None

Returns

<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

Where

<ecode>::<NR1> is the error/event code (-32768 to 32767).

<edesc>::<string> is the description on the error/event.

<einfo>::<string> is the additional information on the error/event. If there is no error, the response is 0, "No events to report".

Examples

STATUS:OFDM:PERROR:EVENTS? might return 12026,"Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually.

STATus:OFDM:POWer:EVENts? (Query Only)

Returns the current events and status condition for the OFDM Power measurement.

Conditions

Measurement views: OFDM

Group

Status commands

Syntax

```
STATus:OFDM:POWer:EVENts?
```

Arguments

None

Returns

```
<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}
```

Where

`<ecode>::<NR1>` is the error/event code (-32768 to 32767).

`<edesc>::<string>` is the description on the error/event.

`<einfo>::<string>` is the additional information on the error/event. If there is no error, the response is 0, "No events to report".

Examples

`STATUS:OFDM:POWER:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:OFDM:STABle:EVENts? (Query Only)

Returns the current events and status condition for the OFDM Symbol table measurement.

Conditions

Measurement views: OFDM

Group

Status commands

Syntax

```
STATus:OFDM:STABle:EVENts?
```

Arguments

None

Returns

```
<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}
```

Where

`<ecode>::<NR1>` is the error/event code (-32768 to 32767).

`<edesc>::<string>` is the description on the error/event.

`<einfo>::<string>` is the additional information on the error/event. If there is no error, the response is 0, "No events to report".

Examples

`STATUS:OFDM:STABLE:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:OPERation:CONDition? (Query Only)

Returns the contents of the Operation Condition Register (OCR).

Conditions

Measurement views: All

Group

Status commands

Syntax

```
STATus:OPERation:CONDition?
```

Arguments

None

Returns

<NR1> is a decimal number showing the contents of the OCR.

Examples

STATUS:OPERATION:CONDITION? might return 16, showing that the bits in the OCR have the binary value 0000000000010000, which means the analyzer is in measurement.

STATus:OPERation:ENABle

Sets or queries the enable mask of the Operation Enable Register (OENR) which allows true conditions in the Operation Event Register to be reported in the summary bit.

Conditions

Measurement views: All

Group

Status commands

Syntax

```
STATus:OPERation:ENABle <bit_value>
STATus:OPERation:ENABle?
```

Arguments

<bit_value>::<NR1> is the enable mask of the OENR. Range: 0 to 65535.

Returns

<NR1> is a decimal number showing the contents of the OENR. Range: 0 to 32767 (The most-significant bit cannot be set true.)

Examples

STATUS:OPERATION:ENABLE1 enables the ALIGning bit.

STATUS:OPERATION:ENABLE? might return 1, showing that the bits in the OENR have the binary value 00000000 00000001, which means that the ALIGning bit is valid.

STATus:OPERation[:EVENt]? (Query Only)

Returns the contents of the Operation Event Register (OEVR). Reading the OEVR clears it.

Conditions

Measurement views: All

Group

Status commands

Syntax

```
STATus:OPERation[:EVENt]?
```

Arguments

None

Returns

<NR1> is a decimal number showing the contents of the OEVR.

Examples

STATUS:OPERATION:EVENT? might return 1, showing that the bits in the OEVR have the binary value 00000000 00000001, which means that the ALIGning bit is set.

STATus:OPERation:NTRansition

Sets or queries the negative transition filter value of the Operation Transition Register (OTR).

Conditions

Measurement views: All

Group

Status commands

Syntax

```
STATus:OPERation:NTRansition <bit_value>
STATus:OPERation:NTRansition?
```

Arguments

<bit_value>::<NR1> is the negative transition filter value. Range: 0 to 65535.

Returns

<NR1> is a decimal number showing the contents of the OTR. Range: 0 to 32767 (The most-significant bit cannot be set true.)

Examples

STATUS:OPERATION:NTRANSITION#H0011 sets the negative transition filter value to #H0011.

STATUS:OPERATION:NTRANSITION? might return 17.

STATus:OPERation:PTRansition

Sets or queries the positive transition filter value of the Operation Transition Register (OTR).

Conditions

Measurement views: All

Group

Status commands

Syntax

```
STATus:OPERation:PTRansition <bit_value>
STATus:OPERation:PTRansition?
```

Arguments

`<bit_value>::<NR1>` is the positive transition filter value. Range: 0 to 65535.

Returns

`<NR1>` is a decimal number showing the contents of the OTR. Range: 0 to 32767 (The most-significant bit cannot be set true.)

Examples

`STATUS:OPERATION:PTRANSITION 0` sets the positive transition filter value to 0.

`STATUS:OPERATION:PTRANSITION?` might return `0`.

STATus:P25:EDIagram:EVENts? (Query Only)

Returns the current events and status conditions for the P25 Eye Diagram display.

Conditions

Measurement view: P25 Eye Diagram

Group

Status commands

Syntax

```
STATus:P25:EDIagram:EVENts?
```

Returns

`<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where:

`<ecode>::<NR1>` is the error/event code (-32768 to 32767).

`<edesc>::<string>` is the description of the error/event.

`<einfo>::<string>` is the additional information on the error/event. If there is no error, the response is 0, "No events to report".

Examples

`STATUS:P25:EDIAGRAM:EVENTS?` might return `12026,"Acq Sampling Parma's: Manual control"`, indicating that the sampling parameters are controlled Manually.

STATus:P25:SUMMary:EVENts (Query Only)

Returns the current events and status conditions for the Summary measurement.

Conditions

Measurement views: P25 Summary

This command requires P25 Measurements.

Group

Status commands

Syntax

`STATus:P25:SUMMary:EVENts`

Arguments

None

Returns

`<ecode>,"<edesc>[<einfo]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where

`<ecode>::<NR1>` is the error/event code (-32768 to 32767).

`<edesc>::<string>` is the description of the error/event.

`<einfo>::<string>` is the additional information on the error/event. If there is no error, then response is 0, “No events to report.”

Examples

`STATUS:P25:SUMMARY:EVENTS?` might return `12026 "Acquisition Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:PERRor[:MEASview<y>]:EVENts? (Query only)

Returns the current events and status conditions for the Phase error versus Time measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Phase error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Phase error versus Time).

Group

Status commands

Syntax

`STATus:PERRor[:MEASview<y>]:EVENts?`

Arguments

None

Returns

<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

Where

<ecode>::<NR1> is the error or event code (-32768 to 32767).

<edesc>::<string> is the description on the error or event.

<einfo>::<string> is the additional information on the error or event. If there is no error, the response is 0, "No events to report".

Examples

STATus:PERRor:MEASview2:EVENts? might return 12026, "Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually for measurement view 2.

STATus:PHVTime[:MEASview<y>]:EVENts? (Query Only)

Returns the current events and status conditions for the Phase versus Time measurement.

Conditions

Measurement views: Phase versus Time

Group

Status commands

Syntax

STATus:PHVTime[:MEASview<y>]:EVENts?

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

Where

<ecode>::<NR1> is the error/event code (-32768 to 32767).

<edesc>::<string> is the description on the error/event.

<einfo>::<string> is the additional information on the error/event. If there is no error, the response is 0, "No events to report".

Examples

STATUS:PHVTIME:MEASview1:EVENTS? might return 12026,"Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually.

STATus:PNOise:EVENts? (Query Only)

Returns the current events and status conditions for the phase noise measurement.

Conditions

Measurement views: Spurious

Group

Status commands

Syntax

`STATus:PNOise:EVENts?`

Arguments

None

Returns

`<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where

`<ecode>::<NR1>` is the error/event code (-32768 to 32767).

`<edesc>::<string>` is the description on the error/event.

`<einfo>::<string>` is the additional information on the error/event. If there is no error, the response is 0, "No events to report".

Examples

`STATUS:PNOISE:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:PRESet (No Query Form)

Presets the SCPI enable registers and transition registers.

Conditions

Measurement views: All

Group

Status commands

Syntax

`STATus:PRESet`

Arguments

None

Examples

`STATUS:PRESET` presets the SCPI enable registers and transition registers.

STATus:PULSe:CUMulative:HISTogram[:MEASview<y>]:EVENts (Query Only)

Returns the current events and status conditions for the Cumulative Histogram display.

Conditions

Measurement views: Cumulative Histogram with the specified Measurement View Number

Group

Status commands

Syntax

`STATus:PULSe:CUMulative:HISTogram[:MEASview<y>]:EVENts?`

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<ecode>,"<edesc>[<einfo]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where

`<ecode>::<NR1>` is the error/event code (-32768 to 32767).

`<edesc>::<string>` is the description of the error/event.

`<einfo>::<string>` is the additional information on the error/event. If there is no error, then response is 0, “No events to report.”

Examples

`STATus:PULSe:CUMulative:HISTogram:MEASview1:EVENTS?` might return `0,"No events to report"`.

STATus:PULSe:CUMulative:STATistics[:MEASview<y>]:EVENts (Query Only)

Returns the current events and status conditions for the Cumulative Statistics display.

Conditions

Measurement views: Pulse Cumulative Statistics

Group

Status commands

Syntax

`STATus:PULSe:CUMulative:STATistics[:MEASview<y>]:EVENts?`

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Returns

`<ecode>,"<edesc>[<einfo]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where

`<ecode>::<NR1>` is the error/event code (-32768 to 32767).

`<edesc>::<string>` is the description of the error/event.

`<einfo>::<string>` is the additional information on the error/event. If there is no error, then response is 0, “No events to report.”

Examples

`STATus:PULSe:CUMulative:STATistics:MEASview1:EVENts?` might return `0,"No events to report"`.

STATus:PULSe:OGRAm[:MEASview<y>]:EVENts (Query Only)

Returns the current events and status conditions for the Pulse-Ogram display.

Conditions

Measurement views: Pulse-Ogram

Group

Status commands

Syntax

`STATus:PULSe:OGRAm[:MEASview<y>]:EVENts?`

Arguments

`<y>` = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Returns

`<ecode>,"<edesc>[<einfo]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where

`<ecode>::<NR1>` is the error/event code (-32768 to 32767).

`<edesc>::<string>` is the description of the error/event.

`<einfo>::<string>` is the additional information on the error/event. If there is no error, then response is 0, “No events to report.”

Examples

`STATus:PULSe:OGRAm:MEASview1:EVENts?` might return `0,"No events to report"`.

STATus:PULSe:RESult:EVENts? (Query Only)

Returns the current events and status conditions for the pulse table measurement.

Conditions

Measurement views: Pulse table

Group

Status commands

Syntax

`STATus:PULSe:RESult:EVENts?`

Arguments

None

Returns

<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

Where

<ecode>::<NR1> is the error/event code (-32768 to 32767).

<edesc>::<string> is the description on the error/event.

<einfo>::<string> is the additional information on the error/event. If there is no error, the response is 0, "No events to report".

Examples

STATUS:PULSE:RESULT:EVENTS? might return 12026,"Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually.

STATus:PULSe:STATistics[:MEASview<y>]:EVENts? (Query Only)

Returns the current events and status conditions for the pulse statistics measurement.

Conditions

Measurement views: Pulse statistics

Group

Status commands

Syntax

STATus:PULSe:STATistics[:MEASview<y>]:EVENts?

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Returns

<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

Where

<ecode>::<NR1> is the error/event code (-32768 to 32767).

<edesc>::<string> is the description on the error/event.

<einfo>::<string> is the additional information on the error/event. If there is no error, the response is 0, "No events to report".

Examples

STATus:PULSe:STATistics:MEASview1:EVENts? might return 12026,"Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually.

STATus:PULSe:TRACe[:MEASview<y>]:EVENts

Returns the current events and status conditions for the pulse trace measurement.

Conditions

Measurement views: Pulse trace with specified Measurement view number

Group

Status commands

Syntax

```
STATus:PULSe:TRACe[:MEASview<y>]:EVENts?
```

Arguments

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

```
<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}
```

Where

<ecode>::<NR1> is the error/event code (-32768 to 32767).

<edesc>::<string> is the description on the error/event.

<einfo>::<string> is the additional information on the error/event. If there is no error, the response is 0, "No events to report".

Examples

`STATUS:PULSE:TRACE:MEASVIEW1:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually on pulse trace measurement view 1.

STATus:QUEStionable:CALibration:CONDition? (Query Only)

Returns the contents of the questionable calibration condition register.

Conditions

Measurement views: All

Group

Status commands

Syntax

```
STATus:QUEStionable:CALibration:CONDition?
```

Arguments

None

Returns

<NR1> is a decimal number showing the contents of the questionable calibration condition register.

Examples

`STATUS:QUESTIONABLE:CALIBRATION:CONDITION?` might return `16384`, showing that the bits in the questionable calibration condition register have the binary value 01000000 00000000, which means the Alignment Needed bit is set.

STATus:QUEStionable:CALibration:ENABle

Sets or queries the enable mask of the questionable calibration enable register which allows true conditions in the questionable calibration event register to be reported in the summary bit.

Conditions

Measurement views: All

Group

Status commands

Syntax

```
STATus:QUEStionable:CALibration:ENABle <bit_value>
STATus:QUEStionable:CALibration:ENABle?
```

Arguments

<bit_value>::<NR1> is the enable mask of the questionable calibration enable register. Range: 0 to 65535.

Returns

<NR1> is a decimal number showing the contents of the questionable calibration enable register. Range: 0 to 32767 (The most-significant bit cannot be set true.)

Examples

STATUS:QUESTIONABLE:CALIBRATION:ENABLE 16384 enables the Alignment Needed bit.

STATUS:QUESTIONABLE:CALIBRATION:ENABLE? might return 16384, showing that the bits in the questionable calibration enable register have the binary value 01000000 00000000, which means that the Calibration Summary bit is valid.

STATus:QUEStionable:CALibration[:EVENt]? (Query Only)

Returns the contents of the questionable calibration event register. Reading the register clears it.

Conditions

Measurement views: All

Group

Status commands

Syntax

```
STATus:QUEStionable:CALibration[:EVENt]?
```

Arguments

None

Returns

<NR1> is a decimal number showing the contents of the questionable calibration event register.

Examples

STATUS:QUESTIONABLE:CALIBRATION:EVENT? might return 16384, showing that the bits in the questionable calibration event register have the binary value 01000000 00000000, which means that the Calibration Summary bit is set.

STATus:QUEStionable:CALibration:NTRansition

Sets or queries the negative transition filter value of the questionable calibration transition register.

Conditions

Measurement views: All

Group

Status commands

Syntax

```
STATus:QUEStionable:CALibration:NTRansition <bit_value>
STATus:QUEStionable:CALibration:NTRansition?
```

Arguments

`<bit_value>::<NR1>` is the negative transition filter value. Range: 0 to 65535.

Returns

<NR1> is a decimal number showing the contents of the questionable calibration transition register. Range: 0 to 32767 (The most-significant bit cannot be set true.)

Examples

`STATUS:QUESTIONABLE:CALIBRATION:NTRANSITION #H4000` sets the negative transition filter value to #H4000.

`STATUS:QUESTIONABLE:CALIBRATION:NTRANSITION?` might return `16384`.

STATus:QUEStionable:CALibration:PTRansition

Sets or queries the positive transition filter value of the questionable calibration transition register.

Conditions

Measurement views: All

Group

Status commands

Syntax

```
STATus:QUEStionable:CALibration:PTRansition <bit_value>
STATus:QUEStionable:CALibration:PTRansition?
```

Arguments

`<bit_value>::<NR1>` is the positive transition filter value. Range: 0 to 65535.

Returns

`<NR1>` is a decimal number showing the contents of the questionable calibration transition register. Range: 0 to 32767 (The most-significant bit cannot be set true.)

Examples

`STATUS:QUESTIONABLE:CALIBRATION:PTRANSITION 0` sets the positive transition filter value to 0.

`STATUS:QUESTIONABLE:CALIBRATION:PTRANSITION?` might return `0`.

STATus:QUEStionable:CONDition? (Query Only)

Returns the contents of the Questionable Condition Register (QCR).

Conditions

Measurement views: All

Group

Status commands

Syntax

```
STATus:QUEStionable:CONDition?
```

Arguments

None

Returns

<NR1> is a decimal number showing the contents of the QCR.

Examples

STATUS:QUESTIONABLE:CONDITION? might return 256, showing that the bits in the QCR have the binary value 00000001 00000000, which means the Calibration Summary bit is set.

STATus:QUEStionable:ENABle

Sets or queries the enable mask of the Questionable Enable Register (QENR) which allows true conditions in the Questionable Event Register to be reported in the summary bit.

Conditions

Measurement views: All

Group

Status commands

Syntax

```
STATus:QUEStionable:ENABle <bit_value>
STATus:QUEStionable:ENABle?
```

Arguments

<bit_value>::<NR1> is the enable mask of QENR. Range: 0 to 65535.

Returns

<NR1> is a decimal number showing the contents of the QENR. Range: 0 to 32767 (The most-significant bit cannot be set true.)

Examples

STATUS:QUESTIONABLE:ENABLE 256 enables the Calibration Summary bit.

STATUS:QUESTIONABLE:ENABLE? might return 256, showing that the bits in the QENR have the binary value 00000001 00000000, which means that the Calibration Summary bit is valid.

STATus:QUEStionable[:EVENt]? (Query Only)

Returns the contents of the Questionable Event Register (QEVR). Reading the QEVR clears it.

Conditions

Measurement views: All

Group

Status commands

Syntax

```
STATus:QUEStionable[:EVENt]?
```

Arguments

None

Returns

<NR1> is a decimal number showing the contents of the QEVR.

Examples

STATUS:QUESTIONABLE:EVENT? might return 256, showing that the bits in the QEVR have the binary value 00000001 00000000, which means that the Calibration Summary bit is set.

STATus:QUEStionable:FREQuency:CONDition? (Query Only)

Returns the contents of the questionable frequency condition register.

Conditions

Measurement views: All

Group

Status commands

Syntax

```
STATus:QUEStionable:FREQuency:CONDition?
```

Arguments

None

Returns

<NR1> is a decimal number showing the contents of the questionable frequency condition register.

Examples

STATUS:QUESTIONABLE:FREQUENCY:CONDITION? might return 512, showing that the bits in the questionable frequency condition register have the binary value 00000010 00000000, which means the Locked To External Ref bit is set.

STATus:QUEStionable:FREQuency:ENABle

Sets or queries the enable mask of the questionable frequency enable register which allows true conditions in the questionable frequency event register to be reported in the summary bit.

Conditions

Measurement views: All

Group

Status commands

Syntax

```
STATus:QUEStionable:FREQuency:ENABle <bit_value>
STATus:QUEStionable:FREQuency:ENABle?
```

Arguments

<bit_value>::<NR1> is the enable mask of the questionable frequency enable register. Range: 0 to 65535.

Returns

<NR1> is a decimal number showing the contents of the questionable frequency enable register. Range: 0 to 32767 (The most-significant bit cannot be set true.)

Examples

STATUS:QUESTIONABLE:FREQUENCY:ENABLE 512 enables the Locked To External Ref bit.

STATUS:QUESTIONABLE:FREQUENCY:ENABLE? might return 512, showing that the bits in the questionable calibration enable register have the binary value 00000010 00000000, which means that the Locked To External Ref bit is valid.

STATus:QUEStionable:FREQuency[:EVENt]? (Query Only)

Returns the contents of the questionable frequency event register. Reading the register clears it.

Conditions

Measurement views: All

Group

Status commands

Syntax

```
STATus:QUEStionable:FREQuency[:EVENt]?
```

Arguments

None

Returns

<NR1> is a decimal number showing the contents of the questionable frequency event register.

Examples

STATUS:QUESTIONABLE:FREQUENCY:EVENT? might return 512, showing that the bits in the questionable frequency event register have the binary value 00000010 00000000, which means that the Locked To External Ref bit is set.

STATus:QUEStionable:FREQuency:NTRansition

Sets or queries the negative transition filter value of the questionable frequency transition register.

Conditions

Measurement views: All

Group

Status commands

Syntax

```
STATus:QUEStionable:FREQuency:NTRansition <bit_value>
STATus:QUEStionable:FREQuency:NTRansition?
```

Arguments

<bit_value>::<NR1> is the negative transition filter value. Range: 0 to 65535.

Returns

<NR1> is a decimal number showing the contents of the questionable frequency transition register. Range: 0 to 32767 (The most-significant bit cannot be set true.)

Examples

STATUS:QUESTIONABLE:FREQUENCY:NTRANSITION #H0200 sets the negative transition filter value to #H0200.

STATUS:QUESTIONABLE:FREQUENCY:NTRANSITION? might return 512.

STATus:QUEStionable:FREQuency:PTRansition

Sets or queries the positive transition filter value of the questionable frequency transition register.

Conditions

Measurement views: All

Group

Status commands

Syntax

```
STATus:QUEStionable:FREQuency:PTRansition <bit_value>
STATus:QUEStionable:FREQuency:PTRansition?
```

Arguments

<bit_value>::<NR1> is the positive transition filter value. Range: 0 to 65535.

Returns

<NR1> is a decimal number showing the contents of the questionable frequency transition register. Range: 0 to 32767 (The most-significant bit cannot be set true.)

Examples

STATUS:QUESTIONABLE:FREQUENCY:PTRANSITION 0 sets the positive transition filter value to 0.

STATUS:QUESTIONABLE:FREQUENCY:PTRANSITION? might return 0.

STATus:QUEStionable:NTRansition

Sets or queries the negative transition filter value of the Questionable Transition Register (QTR).

Conditions

Measurement views: All

Group

Status commands

Syntax

```
STATus:QUEStionable:NTRansition <bit_value>
STATus:QUEStionable:NTRansition?
```

Arguments

`<bit_value>::<NR1>` is the negative transition filter value. Range: 0 to 65535.

Returns

`<NR1>` is a decimal number showing the contents of the QTR. Range: 0 to 32767 (The most-significant bit cannot be set true.)

Examples

`STATUS:QUESTIONABLE:NTRANSITION #H0020` sets the negative transition filter value to #H0020.

`STATUS:QUESTIONABLE:NTRANSITION?` might return `32`.

STATus:QUEStionable:PTRansition

Sets or queries the positive transition filter value of the Questionable Transition Register (QTR).

Conditions

Measurement views: All

Group

Status commands

Syntax

```
STATus:QUEStionable:PTRansition <bit_value>
STATus:QUEStionable:PTRansition?
```

Arguments

`<bit_value>::<NR1>` is the positive transition filter value. Range: 0 to 65535.

Returns

`<NR1>` is a decimal number showing the contents of the QTR. Range: 0 to 32767 (The most-significant bit cannot be set true.)

Examples

`STATUS:QUESTIONABLE:PTRANSITION 0` sets the positive transition filter value to 0.

`STATUS:QUESTIONABLE:PTRANSITION?` might return `0`.

STATus:SEM:EVENts? (Query Only)

Returns the current events and status conditions for the Spectral Emissions Mask measurement.

Conditions

Measurement views: Spectral Emissions Mask

Group

Status commands

Syntax

`STATus:SEM:EVENts?`

Arguments

None

Returns

`<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where

`<ecode>::<NR1>` is the error/event code (-32768 to 32767).

`<edesc>::<string>` is the description on the error/event.

`<einfo>::<string>` is the additional information on the error/event. If there is no error, the response is 0, "No events to report".

Examples

`STATUS:SEM:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:SGRAM[:MEASview<y>]:EVENts (Query Only)

Returns the current events and status conditions for the spectrogram measurement.

Conditions

Measurement views: Spectrogram

Group

Status commands

Syntax

`STATus:SGRAM[:MEASview<y>]:EVENts?`

Arguments

`<y>` = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

Returns

`<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where

`<ecode>::<NR1>` is the error/event code (-32768 to 32767).

`<edesc>::<string>` is the description on the error/event.

`<einfo>::<string>` is the additional information on the error/event. If there is no error, the response is 0, "No events to report".

Examples

`STATus:SGRAM:MEASview1:EVENts?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:SPECtrum[:MEASview<y>]:EVENts? (Query Only)

Returns the current events and status conditions and status conditions for the spectrum measurement.

Conditions

Measurement views: Spectrum

Group

Status commands

Syntax

`STATus:SPECtrum[:MEASview<y>]:EVENts?`

Arguments

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

`<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where

`<ecode>::<NR1>` is the error/event code (-32768 to 32767).

`<edesc>::<string>` is the description on the error/event.

`<einfo>::<string>` is the additional information on the error/event. If there is no error, the response is 0, "No events to report".

Examples

`STATUS:SPECTRUM:MEASVIEW2:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:SPURious:EVENts ? (Query Only)

Returns the current events and status conditions for the Spurious measurement.

Conditions

Measurement view: Spurious Display

Group

Status commands

Syntax

`STATus:SPURious:EVENts ?`

Arguments

None

Returns

`<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

where

<ecode>:: <NR1> is the error/event code (-32768 to 32767).

<edesc>:: <String> is the description of the error/event.

<einfo>:: <String> is the additional information on the error/event If there is no error, then response is 0, “No events to report.”

Examples

STATUS:SPURIOUS:EVENTS? might return 12026,"Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually.

STATus:SQUality[:MEASview<y>]:EVENts? (Query only)

Returns the current events and status conditions for the signal quality measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Signal quality

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Signal quality).

Group

Status commands

Syntax

STATus:SQUality[:MEASview<y>]:EVENts?

Arguments

None

Returns

<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

Where

<ecode>::<NR1> is the error/event code (-32768 to 32767).

<edesc>::<string> is the description on the error/event.

<einfo>::<string> is the additional information on the error/event. If there is no error, the response is 0, "No events to report".

Examples

STATus:SQUality:MEASview1:EVENts? might return 12026,"Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually for measurement view1.

STATus:TDIagram[:MEASview<y>]:EVENts? (Query only)

Returns the current events and status conditions for the trellis diagram measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Trellis diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Trellis diagram).

Group

Status commands

Syntax

`STATus:TDIagram[:MEASview<y>]:EVENts?`

Arguments

None

Returns

`<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where

`<ecode>::<NR1>` is the error or event code (-32768 to 32767).

`<edesc>::<string>` is the description on the error or event.

`<einfo>::<string>` is the additional information on the error or event. If there is no error, the response is 0, "No events to report".

Examples

`STATus:TDIagram:MEASview2:EVENts?` might return 12026, "Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually.

STATus:TXGain:EVENts? (Query Only)

Queries events for the Transmission Gain display.

Conditions

Measurement view: Transmission Gain

Group

Status commands

Syntax

`STATus:TXGain:EVENts?`

Returns

Examples

`STATUS:TXGAIN:EVENTS?`

STATus:WLAN:AD:CONSte:EVENts? (Query Only)

Queries the current events and status conditions for the Constellation measurement display.

Conditions

Measurement view: 802.11ad Constellation

Group

Status commands

Syntax

`STATus:WLAN:AD:CONSte:EVENts?`

Arguments

None

Returns

`<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

where

`<ecode>:: <NR1>` is the error/event code (-32768 to 32767).

`<edesc>:: <String>` is the description of the error/event.

`<einfo>:: <String>` is the additional information on the error/event If there is no error, then response is 0, “No events to report.”

Examples

`STATUS:WLAN:AD:CONSTE:EVENTS?` might return 12056,"Warm-up period", indicating the error code is 12056 and the description of that error code is “Warm up period”.

STATus:WLAN:AD:EVM:EVENts? (Query Only)

Returns the current events and status conditions for the EVM vs Time measurement.

Conditions

Measurement views: 802.11ad EVM vs Time

Group

Status commands

Syntax

`STATus:WLAN:AD:EVM:EVENts?`

Arguments

None

Returns

`<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

where

`<ecode>:: <NR1>` is the error/event code (-32768 to 32767).

`<edesc>:: <String>` is the description of the error/event.

`<einfo>:: <String>` is the additional information on the error/event If there is no error, then response is 0, “No events to report.”

Examples

STATUS:WLAN:AD:EVM:EVENTS? might return 12026,"Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually.

STATus:WLAN:AD:SUMMary:EVENts (Query Only)

Returns the current events and status conditions for the 802.11ad Summary display.

Conditions

Measurement view: 802.11ad Summary

Group

Status commands

Syntax

STATus:WLAN:AD:SUMMary:EVENts

Arguments

None

Returns

<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

where

<ecode>:: <NR1> is the error/event code (-32768 to 32767).

<edesc>:: <String> is the description of the error/event.

<einfo>:: <String> is the additional information on the error/event If there is no error, then response is 0, “No events to report.”

Examples

STATUS:WLAN:AD:SUMMARY:EVENTS? might return 12026,"Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually.

STATus:WLAN:AY:CONSte:EVENts? (Query Only)

Query the current Events and Status conditions in the Constellation display.

Conditions

Measurement view: 802.11ay Analysis Constellation

Group

Status commands

Syntax

STATus:WLAN:AY:CONSte:EVENts?

Returns

<ecode>,"<edesc>[<einfo>]",{<ecode>,"<edesc>[:<einfo>]"}

where:

<ecode>::=<NR1> is the error/event code (Range: -32768 to 32767)

<edesc>::=<string> is the description of the error/event.

<einfo>::=<string> is the additional information on the error/event. If there are no errors, then the response will be 0, “No events to report.”

Examples

STAT:WLAN:AY:CONS:EVEN? may return 12026, "Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually.

STATus:WLAN:AY:EVM:EVENts? (Query Only)

Query the current events and status conditions for the EVM versus time measurement.

Conditions

Measurement views: 802.11ay Analysis EVM

Group

Status commands

Syntax

STATus:WLAN:AY:EVM:EVENts?

Returns

<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

where:

<ecode>::=<NR1> is the error/event code (-32768 to 32767).

<edesc>::= <string> is the description of the error/event.

<einfo>::=<string> is the additional information on the error/event. If there are no errors, the response is 0, “No events to report”.

Examples

STAT:WLAN:AY:EVM:EVEN? may return 12026, "Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually.

STATus:WLAN:AY:SUMMary:EVENts? (Query Only)

Query the current Events and Status conditions for the Summary measurement view.

Conditions

Measurement view: 802.11ay Analysis Summary

Group

Status commands

Syntax

STATus:WLAN:AY:SUMMary:EVENts?

Returns

<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

where:

<ecode>::=<NR1> is the Error/Event Code. Range: -32768 to 32767.

<edesc>::=<string> is the Description of the Error/Event.

<einfo>::=<string> is the additional information about the Error/Event. If there are no errors, then the response is 0, “No events to report”.

Examples

STAT:WLAN:AY:SUMM:EVEN? may return 12026, "Acq Sampling Params: manual control", indicating that sampling parameters are controlled manually.

STATus:WLAN:CONSte:EVENts? (Query Only)

Returns the current instrument status concatenated with WLAN constellation-specific event information.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Status commands

Syntax

STATus:WLAN:CONSte:EVENts?

Returns

ASCII string representation of instrument status as follows:

<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

Where

<ecode>::<NR1> is the error/event code (-32768 to 32767).

<edesc>::<string> is the description on the error/event.

<einfo>::<string> is the additional information on the error/event. If there is no error, the response is 0, "No events to report".

Examples

STATUS:WLAN:CONSTE:EVENTS? might return 12026,"Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually.

STATus:WLAN:CRESponse:EVENts? (Query Only)

Returns the current events and status conditions for the WLAN Channel Response display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Status commands

Syntax

STATus:WLAN:CRESponse:EVENts?

Returns

<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

Where

<ecode>::<NR1> is the error/event code (-32768 to 32767).

<edesc>::<string> is the description on the error/event.

<einfo>::<string> is the additional information on the error/event. If there is no error, the response is 0, "No events to report".

Examples

STATUS:WLAN:CRES:EVENTS? might return 12026,"Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually.

STATus:WLAN:EVM:EVENts? (Query Only)

Returns the current instrument status concatenated with WLAN EVM-specific event information.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Status commands

Syntax

STATus:WLAN:EVM:EVENts?

Returns

ASCII string representation of instrument status as follows:

<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

Where

<ecode>::<NR1> is the error/event code (-32768 to 32767).

<edesc>::<string> is the description on the error/event.

<einfo>::<string> is the additional information on the error/event. If there is no error, the response is 0, "No events to report".

Examples

STATUS:WLAN:EVM:EVENTS? might return 12026,"Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually.

STATus:WLAN:MERRor:EVENts? (Query Only)

Returns the current instrument status concatenated with WLAN Magnitude Error-specific event information.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Status commands

Syntax

`STATus:WLAN:MERRor:EVENts?`

Returns

`<ecode,"<edesc[<einfo]"{,<ecode,"<edesc[:<einfo]"}`

Where:

`<ecode:: <NR1>` is the error/event code (-32768 to 32767).

`<edesc:: <string>` is the description on the error/event.

`<einfo:: <string>` is the additional information on the error/event. If there is no error, the response is 0, "No events to report".

Examples

`STATUS:WLAN:MERR:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:WLAN:PERRor:EVENts? (Query Only)

Returns instrument status concatenated with WLAN Phase Error-specific event information.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Status commands

Syntax

`STATus:WLAN:PERRor:EVENts?`

Returns

ASCII string representation of instrument status as follows:

`<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where

`<ecode>::<NR1>` is the error/event code (-32768 to 32767).

`<edesc>::<string>` is the description on the error/event.

`<einfo>::<string>` is the additional information on the error/event. If there is no error, the response is 0, "No events to report".

Examples

`STATUS:WLAN:PERR:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:WLAN:PVTime:EVENts? (Query Only)

Returns the current instrument status concatenated with WLAN Power vs. Time-specific event information.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Status commands

Syntax

ASCII string representation of instrument status as follows:

`<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where

`<ecode>::<NR1>` is the error/event code (-32768 to 32767).

`<edesc>::<string>` is the description on the error/event.

`<einfo>::<string>` is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples

`STATUS:WLAN:PVT:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:WLAN:STABle:EVENts? (Query Only)

Returns instrument status concatenated with WLAN Symbol Table-specific event information.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Status commands

Syntax

`STATus:WLAN:STABle:EVENts?`

Returns

ASCII string representation of instrument status as follows:

`<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where

`<ecode>::<NR1>` is the error/event code (-32768 to 32767).

`<edesc>::<string>` is the description on the error/event.

`<einfo>::<string>` is the additional information on the error/event. If there is no error, the response is 0, "No events to report".

Examples

`STATUS:WLAN:STABLE:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:WLAN:SUMMary:EVENts? (Query Only)

Returns the current events and status conditions for the WLAN summary measurement.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Status commands

Syntax

`STATus:WLAN:SUMMary:EVENts?`

Arguments

None

Returns

`<ecode>, "<edesc>[<einfo>]" {, <ecode>, "<edesc>[<einfo>]"}`

Where

`<ecode>::<NR1>` the error/event code (-32768 to 32767).

`<edesc>::<string>` the description on the error/event.

`<einfo>::<string>` the additional information on the error/event. If there is no error, the response is `0, "No events to report"`.

Returns the contents of the Status Byte Register (SBR) in the status/event reporting structure using the Master Summary Status (MSS) bit. Refer to Status and Events (on page 2740), for the register information.

*STB? (Query Only)

Returns the contents of the Status Byte Register (SBR) in the status/event reporting structure using the Master Summary Status (MSS) bit. Refer to Status and Events (on page 2740) for the register information.

Conditions

Measurement views: All

Group

IEEE common commands

Syntax

`*STB?`

Related Commands

*CLS (on page 664)

*ESE (on page 1059)

*ESR? (on page 1059)

*SRE (on page 2482)

Arguments

None

Returns

<NR1> representing the contents of the SBR as a decimal number.

Examples

*STB? might return 96, indicating that the SBR contains binary 0110 0000.

System commands

SYSTem:ANTenna:AMPLifier? (Query Only)

Queries the status of the antenna amplifier.

Conditions

Requires an RSA306B/500A/600A instrument with an antenna.

Group

System commands

Syntax

SYSTem:ANTenna:AMPLifier?

Returns

1 or ON means amplifier is on.

0 or OFF means amplifier is off.

SYSTem:ANTenna:CAPabilities:REPort? (Query Only)

Queries the selected antenna capabilities for the items.

Conditions

Requires an RSA306B/500A/600A instrument with an antenna.

Group

System commands

Syntax

SYSTem:ANTenna:CAPabilities:REPort?

Returns

ReportsNothing = 1, ReportsAzimuth = 2, ReportsElevation = 4,

ReportsActiveMode = 8, ReportsBandChange = 16, ReportsRollAngle = 32,

ReportsButtonPresses = 64

Examples

`SYSTEM:ANTENNA:CAPABILITIES:REPORT?` might return `126` when the set to Simulator. This is 64+32+16+8+4+2.

`SYSTEM:ANTENNA:CAPABILITIES:REPORT?` might return `6`, which indicates capabilities of azimuth and elevation reporting.

SYSTem:ANTenna:CATalog? (Query Only)

Queries the catalog of antennas. Use the SYSTem:ANTenna:SELected command to select an antenna from the catalog.

Conditions

Requires an RSA306B/500A/600A instrument.

Group

System commands

Syntax

`SYSTem:ANTenna:CATalog?`

Returns

`<string>,<string>` is a list of antennas in the catalog.

Examples

`SYSTEM:ANTENNA:CATALOG?` might return `"Alaris DF-A0047","Simulator"`.

SYSTem:ANTenna:CONNect (No Query Form)

Initiates antenna connection.

Conditions

Requires antenna or antenna simulator mode. Requires an RSA306B/500A/600A instrument.

Group

System commands

Syntax

`SYSTem:ANTenna:CONNect { 1 | 0 }`

Arguments

`1` or `ON` connects antenna.

`0` or `OFF` does not connect antenna.

SYSTem:ANTenna:DATA? (Query Only)

Queries the true north azimuth, uncorrected compass bearing, elevation, and roll values of the antenna.

Conditions

Antenna mode must be set to Simulator to report roll value.

Requires an RSA306B/500A/600A instrument.

Group

System commands

Syntax

```
SYSTem:ANTenna:DATA?
```

Related Commands

SYSTem:ANTenna:SETDATA (on page 2538)

Returns

```
<numeric>::<NR3>
```

```
<numeric>,<numeric>,<numeric>,<numeric>
```

Examples

`SYSTEM:ANTENNA:DATA?` might return `65.6509800000,54.2109800000,37.8086400000`, indicating the true north azimuth is 65.6°, the uncorrected compass bearing is 54.2°, and the elevation is 37.8°.

SYSTem:ANTenna:DECLination? (Query Only)

Queries current calculated declination.

Conditions

Requires an RSA306B/500A/600A instrument with an antenna.

Group

System commands

Syntax

```
SYSTem:ANTenna:DECLination?
```

Returns

<NR2> is the calculated declination.

Examples

`SYSTEM:ANTENNA:DECLINATION?` might return `10.0`, indicating that the calculated declination is 10.0°.

SYSTem:ANTenna:DECLination:USER

Sets or queries the manually entered declination.

Conditions

Requires an RSA306B/500A/600A instrument with an antenna.

Group

System commands

Syntax

```
SYSTem:ANTenna:DECLination:USER <NR2>
SYSTem:ANTenna:DECLination:USER?
```

Related Commands

SYSTem:ANTenna:DECLination:USER:STATe (on page 2535)

Arguments

<NR2> is the manually entered declination.

Returns

<NR2> is the manually entered declination.

Examples

SYSTEM:ANTENNA:DECLINATION:USER? might return 10.0000000000, indicating that the manually forced declination is set to 10.0°.

SYSTem:ANTenna:DECLination:USER:STATe

Sets or queries to force declination.

Conditions

Requires an RSA306B/500A/600A instrument with an antenna.

Group

System commands

Syntax

```
SYSTem:ANTenna:DECLination:USER:STATe { 1 | 0 | ON | OFF }
SYSTem:ANTenna:DECLination:USER:STATe?
```

Related Commands

SYSTem:ANTenna:DECLination:USER (on page 2534)

Arguments

1 or ON forces the declination to a user entered value.

0 or OFF sets the declination to not be forced.

Returns

1 or ON means the declination is set to a manually entered value.

0 or OFF means the declination is not forced.

SYSTem:ANTenna:FREQuency:BAND? (Query Only)

Queries the currently selected frequency band for the antenna, assuming that it has the ability to report back which band is selected.

Conditions

Requires an RSA306B/500A/600A instrument.

Group

System commands

Syntax
SYSTem:ANTenna:FREQuency:BAND?

Returns
It returns 0 for all currently available antennas (and the Simulator), because they do not have the ability to report back which band is selected.

Examples
SYSTEM:ANTENNA:FREQUENCY:BAND?

SYSTem:ANTenna:MANufacturer? (Query Only)
Queries the antenna manufacturer.

Conditions
Requires an RSA306B/500A/600A instrument.

Group
System commands

Syntax
SYSTem:ANTenna:MANufacturer?

Returns
<string>

Examples
SYSTEM:ANTENNA:MANUFACTURER? might return "Alaris".

SYSTem:ANTenna:MODel? (Query Only)
Queries the antenna model.

Conditions
Requires an RSA306B/500A/600A instrument.

Group
System commands

Syntax
SYSTem:ANTenna:MODel?

Returns
<string>

Examples
SYSTEM:ANTENNA:MODEL? might return "DF-A0047".

SYSTem:ANTenna:OPTions? (Query Only)

Queries the antenna options.

Conditions

Requires an RSA306B/500A/600A instrument.

Group

System commands

Syntax

```
SYSTem:ANTenna:OPTions?
```

Returns

```
<string>
```

Examples

`SYSTEM:ANTENNA:OPTIONS?` might return `"None"`.

SYSTem:ANTenna:SELected

Sets or queries the selected antenna.

Conditions

Requires an RSA306B/500A/600A instrument.

Group

System commands

Syntax

```
SYSTem:ANTenna:SELected <string>
SYSTem:ANTenna:SELected?
```

Related Commands

SYSTem:ANTenna:CATalog? (on page 2533)

Arguments

`<string>` is an item from the antenna catalog.

`Alaris DF-A0047` sets analyzer to Alaris antenna. Simulator sets the analyzer to the simulated antenna.

Returns

“None” means the analyzer is set to no antenna.

“Alaris DF-A0047” means the analyzer is set to the Alaris antenna. “Simulator” means the analyzer is set to the simulated antenna.

Examples

`SYSTEM:ANTENNA:SELECTED?` might return `"Alaris DF-A0047"`, indicating that the antenna selection is set to the Alaris DF-A0047.

SYSTem:ANTenna:SERial? (Query Only)

Queries the antenna serial number.

Conditions

Requires an RSA306B/500A/600A instrument.

Group

System commands

Syntax

```
SYSTem:ANTenna:SERial?
```

Returns

```
<string>
```

Examples

`SYSTEM:ANTENNA:SERIAL?` might return `"A22500"`.

SYSTem:ANTenna:SETDATA (No Query Form)

Sets the amplifier, true north azimuth, uncorrected compass bearing, elevation, and roll values of the antenna.

Conditions

Antenna mode must be set to Simulator to report roll value.

Requires an RSA306B/500A/600A instrument.

Group

System commands

Syntax

```
SYSTem:ANTenna:SETDATA <boolean>,<numeric>,<numeric>,<numeric>,<numeric>
```

Related Commands

SYSTem:ANTenna:DATA? (on page 2533)

Arguments

<boolean> is { 1 | 0 | ON | OFF }

`<numeric>::<NR3>`

Examples

`SYSTEM:ANTENNA:SETDATA 1,65.6509800000,54.2109800000,37.8086400000` will set the amplifier to ON (1), the true north azimuth to 65.6°, the uncorrected compass bearing to 54.2°, and the elevation to 37.8°.

SYSTem:BATTery:CCYCle? (Query Only)

Queries the battery charge cycle.

Conditions

Requires an RSA500A series instrument with a battery installed.

Group

System commands

Syntax

`SYSTem:BATTery:CCYCle?`

Returns

Returns the number of charge cycles of the battery.

Examples

`SYSTEM:BATTERY:CCYCLE?` might return `10`, indicating that the battery has been charged 10 times.

SYSTem:BATTery:DATE? (Query Only)

Queries the battery manufactured date.

Conditions

Requires an RSA500A series instrument with a battery installed.

Group

System commands

Syntax

`SYSTem:BATTery:DATE?`

Returns

Returns the date string as yyyy-mm-dd.

Examples

`SYSTEM:BATTERY:DATE?` might return `"2015-10-05"`, indicating the battery manufactured date is October 5, 2015.

SYSTem:BATTery:RCHarge? (Query Only)

Queries the relative charge remaining.

Conditions

Requires an RSA500A series or RSA600A series instrument with a battery installed.

Group

System commands

Syntax

`SYSTem:BATTery:RCHarge?`

Returns

Returns the relative charge remaining in percent.

Examples

`SYSTEM:BATTERY:RCHARGE?` might return `50`, indicating the relative charge remaining is 50%.

SYSTem:BATTery:SERial? (Query Only)

Queries the battery serial number.

Conditions

Requires an RSA500A series instrument with a battery installed.

Group

System commands

Syntax

`SYSTem:BATTery:SERial?`

Returns

Returns the battery serial number.

Examples

`SYSTEM:BATTERY:SERIAL?` might return `1234`, indicating the serial number is 1234.

SYSTem:BATTery:STATus? (Query Only)

Queries the battery status.

Conditions

Requires an RSA500A series instrument with a battery installed.

Group

System commands

Syntax

`SYSTem:BATTery:STATus?`

Returns

`ACFull` means AC present and fully charged

`ACCHarging` means AC present and charging

`ACNBattery` means AC present and no battery

`FULL` means battery fully charged

`EMPTy` means battery fully discharged DISCharging battery discharging ERRor battery error

Examples

`SYSTEM:BATTERY:STATUS?` might return ACNB, indicating no battery is installed.

SYSTem:BATTery:TEMPerature? (Query Only)

Queries the battery temperature.

Conditions

Requires an RSA500A series instrument with a battery installed.

Group

System commands

Syntax

```
SYSTem:BATTery:TEMPerature?
```

Returns

Returns the battery temperature in degrees Celsius.

Examples

`SYSTEM:BATTERY:TEMPERATURE?` might return `15`, indicating that the battery temperature is 15 °C.

SYSTem:COMMunicate:GPIB[:SELF]:ADDRess

Sets or queries the GPIB address of the instrument.

Conditions

Measurement views: All

Group

System commands

Syntax

```
SYSTem:COMMunicate:GPIB[:SELF]:ADDRess <value>
SYSTem:COMMunicate:GPIB[:SELF]:ADDRess?
```

Arguments

`<value>::<NR1>` specifies the GPIB address of the instrument.

`*RST` has no effect on the value.

Examples

`SYSTEM:COMMUNICATE:GPIB:SELF:ADDRESS 18` sets the GPIB address to 18.

SYSTem:DATE

Sets or queries the date (year, month, and day). This command is equivalent to the date setting through the Windows Control Panel.

Conditions

Measurement views: All

Group

System commands

Syntax

```
SYSTem:DATE <year>,<month>,<day>
SYSTem:DATE?
```

Related Commands

SYSTem:TIME (on page 2565)

Arguments

<year>::<NRf> specifies the year (4 digits). Range: 2000 to 2099.

<month>::<NRf> specifies the month. Range: 1 (January) to 12 (December).

<day>::<NRf> specifies the day. Range: 1 to 31. These values are rounded to the nearest integer.

*RST has no effect on the settings.

Examples

SYSTEM:DATE 2008,3,19 sets the internal calendar to March 19, 2008.

SYSTem:ERRor:ALL? (Query Only)

Queries the error/event queue for all the unread items and removes them from the queue. The response is a comma separated list of number, string pairs in FIFO order. For details of the error messages, refer to Error messages and codes (on page 2750).

Conditions

Measurement views: All

Group

System commands

Syntax

SYSTem:ERRor:ALL?

Arguments

None

Returns

<ecode>,"<edesc>[;<einfo>]"{,<ecode>,"<edesc>[;<einfo>]"}

Where

<ecode>::<NR1> is the error/event code (-32768 to 32767).

<edesc>::<string> is the description on the error/event.

<einfo>::<string> is the detail of the error/event.

If the queue is empty, the response is 0, "No error; Queue empty - No events to report".

Examples

SYSTEM:ERROR:ALL? might return -130, "Suffix error; Unrecognized suffix, INPutMLEVel -10dB", indicating that the unit of the reference level is improper.

SYSTem:ERRor:CODE:ALL? (Query Only)

Queries the error/event queue for all the unread items and removes them from the queue. The response returns a comma separated list of only the error/event code numbers in FIFO order. For details of the error messages, refer to Error messages and codes (on page 2750).

Conditions

Measurement views: All

Group

System commands

Syntax

`SYSTem:ERRor:CODE:ALL?`

Arguments

None

Returns

`<ecode>{,<ecode>}`

Where

`<ecode>::<NR1>` is the error/event code, ranging from -32768 to 32767.

If the queue is empty, the response is 0.

Examples

`SYSTEM:ERROR:CODE:ALL?` might return `-101,-108` of the error codes.

SYSTem:ERRor:CODE[:NEXT]? (Query Only)

Queries the error/event queue for the next item and removes it from the queue. The response returns only the error/event code number omitting the string. Except for the shortened response, the query operates identically to SYSTem:ERRor[:NEXT]?. For details of the error messages, refer to Error messages and codes (on page 2750).

Conditions

Measurement views: All

Group

System commands

Syntax

`SYSTem:ERRor:CODE[:NEXT]?`

Arguments

None

Returns

`<ecode>::<NR1>` is the error/event code, ranging from −32768 to 32767.

Examples

`SYSTEM:ERROR:CODE:NEXT?` might return `-101` of the error code.

SYSTem:ERRor:COUNt? (Query Only)

Queries the error/event queue for the number of unread items. As errors and events may occur at any time, more items may be present in the queue at the time it is actually read.

Conditions

Measurement views: All

Group

System commands

Syntax

`SYSTem:ERRor:COUNt?`

Arguments

None

Returns

`<enum>::<NR1>` is the number of errors/events.

If the queue is empty, the response is 0.

Examples

`SYSTEM:ERROR:COUNT?` might return `2`, indicating that the error/event queue contains two of unread errors/events.

SYSTem:ERRor[:NEXT]? (Query Only)

Queries the error/event queue for the next item and removes it from the queue. The response returns the full queue item consisting of an integer and a string. For details of the error messages, refer to the Measurement status conditions table in the PDF of the Programmer manual for your product.

Conditions

Measurement views: All

Group

System commands

Syntax

`SYSTem:ERRor[:NEXT]?`

Arguments

None

Returns

`<ecode>,"<edesc>[;<einfo>]"`

Where

`<ecode>::<NR1>` is the error/event code, ranging from −32768 to 32767.

`<edesc>::<string>` is the description on the error/event.

`<einfo>::<string>` is the additional information on the error/event.

Examples

`SYSTEM:ERROR:NEXT?` might return `-130, "Suffix error; Unrecognized suffix, INPutMLEVel -10dB"`, indicating that the unit is improper.

SYSTem:GNSS:ALTitude (No Query Form)

Query the GNSS altitude.

Conditions

Measurement views: All

Requires an RSA306B/500A/600A instrument.

Group

System commands

Syntax

`SYSTem:GNSS:ALTitude <altitude>`

Returns

`<altitude>::<NRF>` in meters.

Examples

`SYSTEM:GNSS:ALTITUDE` might return `60.1`, the GNSS altitude.

SYSTem:GNSS:COURse? (Query Only)

Query the GNSS course in degrees.

Conditions

Measurement views: All

Requires an RSA306B/500A/600A instrument.

Group

System commands

Syntax

`SYSTem:GNSS:COURse?`

Examples

`SYSTEM:GNSS:COURSE?` might return `162.78`, indicating the GNSS course is 162.78 degrees.

SYSTem:GNSS:DATA? (Query Only)

Query the GNSS data.

Conditions

Measurement views: All

Requires an RSA306B/500A/600A instrument.

Group

System commands

Syntax

`SYSTem:GNSS:DATA?`

Returns

<status>, <longtitude>, <latitude>, <altitude>, <timestamp>, <speed>, <course>, satellites> <HDOP>

Examples

SYSTEM:GNSS:DATA? might return
LOCK,-122.8197333333,45.4994000000,63.4000000000,"2012-06-01T13:51:26.00",22.96480

SYSTem:GNSS:HDOP? (Query Only)

Query the GNSS HDOP value.

Conditions

Measurement views: All

Requires an RSA306B/500A/600A instrument.

Group

System commands

Syntax

SYSTem:GNSS:HDOP?

Returns

<value>::<NR1> the horizontal DOP value.

Examples

SYSTEM:GNSS:HDOP? might return 900.000E-3, the GNSS HDOP value.

SYSTem:GNSS:LATitude? (Query Only)

Query the GNSS latitude.

Conditions

Measurement views: All

Requires an RSA306B/500A/600A instrument.

Group

System commands

Syntax

SYSTem:GNSS:LATitude?

Examples

SYSTEM:GNSS:LATITUDE? might return 45.497193, the GNSS latitude.

SYSTem:GNSS:LONGitude? (Query Only)

Query the GNSS longtitude.

Conditions

Measurement views: All

Requires an RSA306B/500A/600A instrument.

Group

System commands

Syntax

```
SYSTem:GNSS:LONGitude?
```

Examples

`SYSTEM:GNSS:LONGITUDE?` might return the `-122.82372`, the GNSS longitude.

SYSTem:GNSS:SATellites? (Query Only)

Query the number of GNSS satellites.

Conditions

Measurement views: All

Requires an RSA306B/500A/600A instrument.

Group

System commands

Syntax

```
SYSTem:GNSS:SATellites?
```

Examples

`SYSTEM:GNSS:SATELLITES?` might return `10.000`, indicating the number of GNSS satellites is 10.

SYSTem:GNSS:SELected

Selects or queries the source for the GNSS signal.

The INTernal argument is not available with the RSA306.

Conditions

Measurement views: All

Group

System commands

Syntax

```
SYSTem:GNSS:SELected { NONE | INTernal | EXTernal }
SYSTem:GNSS:SELected?
```

Arguments

`NONE`: Selects no GNSS system.

`EXTernal`: This can be an external USB device or other external device on the table.

Returns

`NONE`: No GNSS system is selected.

EXTernal: The external GNSS system.

Examples

SYSTEM:GNSS:SELECTED? might return EXT, indicating the source for the GNSS system is an external device.

SYSTem:GNSS:SERial:BAUD

Sets or queries the GNSS serial baud rate.

Conditions

Measurement views: All

Requires an RSA306B/500A/600A instrument.

Group

System commands

Syntax

```
SYSTem:GNSS:SERial:BAUD { 4800, 9600, 19200, 38400 }
SYSTem:GNSS:SERial:BAUD?
```

Examples

SYSTEM:GNSS:SERIAL:BAUD 19200 sets the baud rate to 19200.

SYSTem:GNSS:SERial:PORT

Sets or queries the GNSS serial communications port.

Conditions

Measurement views: All

Requires an RSA306B/500A/600A instrument.

Group

System commands

Syntax

```
SYSTem:GNSS:SERial:PORT <COM_Port>
SYSTem:GNSS:SERial:PORT?
```

Examples

SYSTEM:GNSS:SERIAL:PORT 1 sets PORT 1 as the GNSS serial communications port.

SYSTem:GNSS:SPEed? (Query Only)

Query the GNSS speed in meter/hour.

Conditions

Measurement views: All

Requires an RSA306B/500A/600A instrument.

Group

System commands

Syntax
`SYSTem:GNSS:SPEed?`

Examples

`SYSTEM:GNSS:SPEED?` might return `31.33583`, indicating the GNSS speed is 31.33583 meters/hour.

SYSTem:GNSS:STATus? (Query Only)

Query the GNSS status.

Conditions

Measurement views: All

Requires an RSA306B/500A/600A instrument.

Group

System commands

Syntax
`SYSTem:GNSS:STATus?`

Returns

`{ OFF | BAD | UNLock | LOCKed }`

Where:

`OFF` - the GNSS is turned off.

`BAD` - a communication error.

`UNL` - the GNSS is unlocked.

`LOCK` - the GNSS is locked.

Examples

`SYSTEM:GNSS:STATUS?` might return `BAD`, indicating a communication error has occurred. For example, the GNSS port or baud rate is wrong.

SYSTem:GNSS:TIMestamp? (Query Only)

Query the GNSS timestamp.

Conditions

Measurement views: All

Requires an RSA306B/500A/600A instrument.

Group

System commands

Syntax
`SYSTem:GNSS:TIMestamp?`

Examples

SYSTEM:GNSS:TIMESTAMP? might return 2012-06-01T13:47:21.00, the GNSS timestamp.

SYSTem:KLOCk

Enables or disables the local lockout operation.

Conditions

Measurement views: All

Group

System commands

Syntax

```
SYSTem:KLOCk { OFF | ON | 0 | 1 }
SYSTem:KLOCk?
```

Arguments

OFF or 0 disables the local lockout operation.

ON or 1 enables the local lockout operation.

Examples

SYSTEM:KLOCK ON enables the local lockout operation.

SYSTem:LINK:SETTings

Sets or queries whether or not global settings linking enabled for the measurements displayed.

Conditions

Measurement views: General Signal viewing and GPDemod Measurements.

Group

System commands

Syntax

```
SYSTem:LINK:SETTings
```

Arguments

0 indicates global settings linking is disabled.

1 indicates global settings linking is enabled.

Examples

SYSTem:LINK:SETTings 1 enables the global settings linking for the displayed measurement views.

SYSTem:LINK:SETTings:SYNC

Sets the synchronize settings referring to source for the measurements displayed.

Conditions

Measurement views: General Signal viewing and GPDemod Measurements.

Group

System commands

```
Syntax
SYSTem:LINK:SETTings:SYNC { 'Source1' | 'Source2' | 'Source3' | 'Source4'}
```

Arguments

`'Source1'` or `'Source2'`.

Examples

`SYSTem:LINK:SETTings:SYNC 'Source2'` synchronizes the settings referring to Source2 for the displayed measurement views.

SYSTem:OPTions? (Query Only)

Queries the options installed in the analyzer. This command is equivalent to the IEEE common command *OPT?.

Conditions

Measurement views: All

Group

System commands

Syntax

```
SYSTem:OPTions?
```

Arguments

None

Returns

`<option>::<string>` contains the comma-separated option numbers.

Examples

`SYSTEM:OPTIONS?` might return `"01,02,20"`, indicating that Option 01, 02, and 20 are currently installed in the analyzer.

SYSTem:PRESet (No Query Form)

Restores the analyzer to the defaults. This command is equivalent to the Preset key on the front panel.

Conditions

Measurement views: All

Group

System commands

Syntax

```
SYSTem:PRESet
```

Arguments

None

Examples

SYSTEM:PRESET restores the analyzer to the defaults.

SYSTem:PRESet:APPLication:ACTion

Sets or queries the preset action for the application preset type.

Conditions

Measurement views: All

Group

System commands

Syntax

```
SYSTem:PRESet:APPLication:ACTion { RECall | SHOW }
SYSTem:PRESet:APPLication:ACTion?
```

Arguments

RECall recalls the selected preset for the application preset type.

SHOW lists the available presets for the application preset type.

Examples

SYSTEM:PRESET:APPLICATION:ACTION SHOW lists the available presets for the application preset type.

SYSTem:PRESet:BLUEtooth:STANdard

Sets or queries the standard, setup, Retaining Current Reference level, and Retaining Center Frequency in the Bluetooth standard preset.

Conditions

Measurement views: Launch BT Application

Group

System commands

Syntax

```
SYSTem:PRESet:BLUEtooth:STANdard {BR | LE | EDR}, {BW20DB | MODD | INBE | NCOM}, {1 | 0}, {1 | 0}
```

Arguments

BR is the Basic Rate.

LR is the Low Energy Rate.

EDR is the Enhanced Data Rate.

<test_setup> :: = {BW20DB|MODD|INBE|NCOM} the name of the test setup.

<Retaining_Current_Reflevel> :: = [1 | 0] where 1 indicates this value is checked in the view; 0 indicates the value is unchecked.

<Retaining_Center_Frequency> :: = [1 | 0] where 1 indicates this value is checked in the view; 0 indicates the value is unchecked.

Examples

`SYSTEM:PRESET:BLUETOOTH:STANDARD BR,MODD,0,1` will set Preset to Bluetooth, standard to basic rate, Test Setup to Modulation/Frequency Offset/Drift/Output Power, disable the retain current center frequency setting, Enable the Retain current ref level setting.

SYSTem:PRESet:DPX (No Query Form)

Restores the analyzer to the defaults for the DPX preset type.

Conditions

Measurement views: All

Group

System commands

```
Syntax
SYSTem:PRESet:DPX { SWEPt | REALtime | ZERospan }
```

Arguments

`SWEPt` restores the setup of the Swept DPX measurement.

`REALtime` restores the setup of the Realtime DPX measurement (110 MHz span).

`ZERospan` restores the setup of the Zero Span DPX measurement.

Examples

`SYSTEM:PRESET:DPX REALtime` restores the DPX, Spectrum to realtime.

SYSTem:PRESet:DPX:ACTion

Sets or queries the preset action for the DPX preset type.

Conditions

Measurement views: All

Group

System commands

Syntax

```
SYSTem:PRESet:DPX:ACTion { RECall | SHOW }
SYSTem:PRESet:DPX:ACTion?
```

Arguments

`RECall` recalls the selected preset for the DPX preset type.

`SHOW` lists the available presets for the DPX preset type.

Examples

`SYSTEM:PRESET:DPX:ACTION SHOW` lists the available presets for the DPX preset type.

SYSTem:PRESet:DPX:SELected

Sets or queries the presets for the DPX preset type.

Conditions

Measurement views: All

Group

System commands

Syntax

```
SYSTem:PRESet:DPX:SELected { OPEn | SWEPt | REALtime | ZERospan }
SYSTem:PRESet:DPX:SELected?
```

Arguments

`OPEn` sets the DPX measurement to Open the DPX display.

`SWEPt` sets the DPX measurement to Swept mode.

`REALtime` sets the DPX measurement to Real-time mode (110 MHz span).

`ZERospan` sets the DPX measurement to Zero Span mode.

Examples

`SYSTEM:PRESET:DPX:SELECTED REALtime` sets the DPX, Spectrum to realtime

SYSTem:PRESet:LTE:STANdard

Sets or queries the following: Preset, Test Setup, frame structure channel bandwidth, Base Station category (for LTE ACLR and SEM test setups), Adjacent channel type (for LTE ACLR test setup), Retain current Center Frequency setting, and Retain current Reference Level.

Conditions

Measurement view: Standards Presets

Group

System commands

Syntax

```
SYSTem:PRESet:LTE:STANdard { MODUlation | ACLR | CS | SEM, FDD | TDD, BW1PT4M | BW3M | BW5M | BW10M
    | BW15M | BW20M, WACA | WACB1 | WACB2 | HACAB | LACAB | MACAB, UTRA | EUTRA,CR1PT28M | CR3PT84M
    | CR7PT68M, 0 | 1, 0 | 1 ]
SYSTem:PRESet:LTE:STANdard?
```

Arguments

`MODUlation | ACLR | CS | SEM, FDD | TDD, BW1PT4M | BW3M | BW5M | BW10M | BW15M | BW20M, WACA | WACB1 | WACB2 | HACAB | LACAB | MACAB, UTRA | EUTRA, CR1PT28M | CR3PT84M | CR7PT68M, 0 | 1, 0 | 1`

Returns

See Arguments.

Examples

`SYSTEM:PRESET:LTE:STANDARD?` might return `MOD,FDD,BW1PT4M,0,1`, indicating that the test setup is Cell ID, Frame Structure, Channel BW, Retain Center Frequency, Retain Reference Level.

SYSTem:PRESet[:MAIN](No Query Form)

Restores the analyzer to the defaults. This command is equivalent to the SYSTem:PRESet command.

Conditions

Measurement views: All

Group

System commands

Syntax

```
SYSTem:PRESet[:MAIN] { MAIN|}
SYSTem:PRESet[:MAIN]?
```

Arguments

`CURRent` restores the analyzer to the current version of the main preset.

`V1` restores the analyzer to the original version of the main preset.

`V2` restores the analyzer to the Full Spectrum version of the main preset.

Examples

`SYSTEM:PRESET[:MAIN] CURRENT` restores the analyzer to the defaults of the current version.

SYSTem:PRESet:MAIN:ACTion

Sets or queries the preset action for the main preset type.

Conditions

Measurement views: All

Group

System commands

```
Syntax
SYSTem:PRESet:MAIN:ACTion { RECall | SHOW }
SYSTem:PRESet:MAIN:ACTion?
```

Arguments

`RECall` recalls the selected preset for the main preset type.

`SHOW` lists the available presets for the main preset type.

Examples

`SYSTEM:PRESET:MAIN:ACTION SHOW` lists the available presets for the main preset type.

SYSTem:PRESet:MAIN:SELected

Sets or queries the presets for the Main preset type.

Conditions

Measurement views: All

Group

System commands

Syntax

```
SYSTem:PRESet:MAIN:SELected { CURRent | V1 | V2 }
SYSTem:PRESet:MAIN:SELected?
```

Arguments

`CURRent` selects the current version of the Main presets.

`V1` selects the previous version of the main preset.

`V2` selects Full Spectrum Sweep version of the current main preset.

Examples

`SYSTEM:PRESET:MAIN:SELECTED CURRent` selects the current version of the Main presets.

SYSTem:PRESet:P25:STANdard (No Query Form)

This command restores the analyzer defaults for the P25 Standards preset.

Conditions

Measurement view: Any P25 measurement

Group

System commands

Syntax

```
SYSTem:PRESet:P25:STANdard <{ PHASe1 | PHASe2 }>,<{ C4FM | HCPM | HDQPSK }>,<{ 1 | 0 }>
```

Arguments

`PHASe1` and `PHASe2` are the P25 standard types.

`C4FM`, `HCPM`, `HDQPSK` are the P25 modulation types.

`1` = retain center frequency; `0` = do not retain center frequency.

Returns

`<standard>` is the P25 standard type: Phase1, Phase2.

`<modulation type>` is the P25 modulation type: C4FM, HCPM (Inbound), HDQPSK (Outbound).

`<{ 1 | 0 }>` controls whether or not the center frequency is retained: 0 = not retained; 1 = retained.

Examples

`SYSTEM:PRESET:P25:STANDARD?` might return `PHASE1,C4FM,0`, indicating that the P25 preset is set to standard type Phase 1, modulation type C4FM, and that the center frequency is set to not be retained.

SYSTem:PRESet:STANdards (No Query Form)

Restores the analyzer to the defaults for the WLAN Standards preset type. To restore the analyzer to a preset file, use the command SYSTem:PRESet:USER (on page 2557).

Conditions

Measurement views: All

Group

System commands

Syntax

```
SYSTem:PRESet:STANdards <WLAN>
```

Arguments

<WLAN> specifies to restore the analyzer to the WLAN standards preset type.

Examples

SYSTEM:PRESET:STANDARDS "WLAN" restores the analyzer to the defaults for the WLAN Standards preset type.

SYSTem:PRESet:STANdards:ACTion

Sets or queries the preset action for the Standards preset type.

Conditions

Measurement views: All

Group

System commands

Syntax

```
SYSTem:PRESet:STANdards:ACTion { RECall | SHOW }
SYSTem:PRESet:STANdards:ACTion?
```

Arguments

RECall recalls the selected preset for the Standards preset type.

SHOW lists the available presets for the Standards preset type.

Examples

SYSTEM:PRESET:STANDARDS:ACTION SHOW lists the available presets for the Standards preset type.

SYSTem:PRESet:USER (No Query Form)

Restores the analyzer to the defaults for the user preset type.

Conditions

Measurement views: All

Group

System commands

Syntax

```
SYSTem:PRESet:USER <file_name>
```

Arguments

<file_name>::<string> specifies the user presets file to restore. The file extension is .Setup. You can omit the extension.

Examples

SYSTEM:PRESET:USER "Spectrum" restores the defaults of the file “Spectrum.Setup” from the C:\RSA5100B Files\User Presets directory.

SYSTem:PRESet:USER:ACTion

Sets or queries the preset action for the user preset type.

Conditions

Measurement views: All

Group

System commands

Syntax

```
SYSTem:PRESet:USER:ACTion { RECall | SHOW }
SYSTem:PRESet:USER:ACTion?
```

Arguments

RECall recalls the selected preset for the user preset type.

SHOW lists the available presets for the user preset type.

Examples

SYSTEM:PRESET:USER:ACTION SHOW lists the available presets for the User preset type.

SYSTem:PRESet:USER:SELected

Sets or queries the presets for the User preset type.

Conditions

Measurement views: All

Group

System commands

Syntax

```
SYSTem:PRESet:USER:SELected <file_name>
SYSTem:PRESet:USER:SELected?
```

Arguments

<file_name>::<string> specifies the User presets file to set. The file extension is .Setup. You can omit the extension.

Examples

SYSTEM:PRESET:USER:SELECTED "Spectrum" sets the User preset to “Spectrum.Setup” saved in C:\RSA5100B Files\User Presets directory.

SYSTem:PRESet:WLAN:AD:STANdard

Sets or queries the 802.11ad preset standard.

Conditions

Measurement views: All 802.11ad measurement views

Group

System commands

Syntax

```
SYSTem:PRESet:WLAN:AD:STANdard {SC| CPHY,SEM| EVM,0| 1,0| 1,CH1| CH2}
SYSTem:PRESet:WLAN:AD:STANdard?
```

Arguments

`SC|CPHY` = Sets standard as either Single carrier or Control PHY

`SEM|EVM` = Sets Test setup as either SEM or EVM

`0|1` = Checks or unchecks the check box for Retain current Center Frequency settings

`0|1` = Checks or unchecks the check box for Retain Current Ref Level setting

`CH1|CH2` = Selects between channel 1 and channel 2

Returns

See Arguments

Examples

`SYSTEM:PRESET:WLAN:AD:STANDARD?` might return `"SC,EVM,0,1,CH2"`, indicating the standard selected is single carrier, test setup selected is EVM, Retain current Center Frequency box is disabled (unchecked), Retain Current Ref Level is enabled (checked), and Channel 2 is selected.

SYSTem:PRESet:WLAN:AY:STANdard

Set or query the WLAN 802.11ay standard, Retain Current Center Frequency setting, and Retain Current Reference Level setting in the WLAN 802.11ay standard preset.

Conditions

Measurement views: All WLAN 802.11ay

Group

System commands

Syntax

```
SYSTem:PRESet:WLAN:AY:STANdard <AYSC>,{ EVM | SEM },{ 0 | 1},{ 0 | 1 },{ CH1 | CH2 },{ BW1X |
    BW2X }
SYSTem:PRESet:WLAN:AY:STANdard?
```

Arguments

Argument	Description
AYSC	Sets the standard to Single Carrier.
EVM | SEM	Sets the test setup as EVM or SEM.
0 | 1	Retains the Center Frequency.

Argument	Description
0 | 1	Retains the Reference Level.
CH1 | CH2	Sets the Radio Frequency Source to Channel 1 or Channel 2.
BW1X | BW2X	Sets the Channel Bandwidth as 2.16 GHz or 4.32 GHz.

Returns

See Arguments.

Examples

`SYST:PRES:WLAN:AY:STAN AYSC,EVM,0,1,CH2,BW1X` sets preset to 802.11ay, standard to Single Carrier, disables the Retain Current Center Frequency setting, enables the Retain Current Reference Level setting, and sets the channel to CH2.

`SYST:PRES:WLAN:AY:STAN?` may return `AYSC, SEM, 1, 0 , CH1, BW2X`, which means that the standard is Single Carrier, the display type is SEM, the Center Frequency is retained, the Reference Level is not retained, Reference Frequency Source is set to channel 1, and the Channel Bandwidth is 4.32 GHz.

SYSTem:PRESet:WLAN:STANdard

Sets or queries the standard and bandwidth presets for the WLAN standards preset type. The first enumeration is the 802.11 standard; the second enumeration is the bandwidth.

Conditions

Measurement views: All

Group

System commands

Syntax

```
SYSTem:PRESet:WLAN:STANdard { B | AC,[BW20|BW40|BW80|BW160] | [A,G],BW20 | P,[BW5|BW10|BW20] | J,
    [BW10|BW20] | N,[BW20|BW40] }
SYSTem:PRESet:WLAN:STANdard?
```

Arguments

`B` selects the 802.11b WLAN standard preset options.

`AC,[BW20|BW40|BW80|BW160]` selects the 802.11b WLAN standard preset options.

`[A,G],BW20` selects the 802.11a or g WLAN standard preset options.

`P,[BW5|BW10|BW20]` selects the 802.11p WLAN standard preset options.

`J,[BW10|BW20]` selects the 802.11j WLAN preset options.

`N,[BW20|BW40]` selects the 802.11n WLAN standard preset options.

Examples

`SYSTEM:PRESET:WLAN:STANDARD P, BW5` selects 802.11p as the WLAN standard to use and 5 MHz as the bandwidth preset.

SYSTem:SVPC:DISconnect (No Query Form)

Disconnects an RSA306B/500A/600A/7100A. If playback is active, exits playback.

Group

System commands

Syntax

```
SYSTem:SVPC:DISconnect
```

Arguments

None.

SYSTem:SVPC:CONNect (No Query Form)

Connects an RSA306B/500A/600A/7100A.

Group

System commands

Syntax

```
SYSTem:SVPC:CONNect
```

Arguments

None.

SYSTem:SVPC:CONNect:RSA300 (No Query Form)

Connects an RSA306 instrument.

Group

System commands

Syntax

```
SYSTem:SVPC:CONNect:RSA300
```

Arguments

None.

SYSTem:SVPC:INSTrument:SERial? (Query Only)

Available when an RSA is connected to SignalVu-PC.

Queries the serial number and model of the connected instrument.

Group

System commands

Syntax

```
SYSTem:SVPC:INSTrument:SERial?
```

Arguments

None.

Returns

Serial number and model of connected instrument.

This commands will not return the correct instrument serial number if an RSA is disconnected and then another connected without using the Search for Instruments tool in the SignalVu-PC application to find and connect to the new instrument.

SYSTem:SVPC:INSTrument:VERSion? (Query Only)

Available when an RSA is connected to SignalVu-PC. Queries the firmware version of the connected instrument.

Group

System commands

Syntax

```
SYSTem:SVPC:INSTrument:VERSion?
```

Arguments

None.

Returns

Firmware version of connected instrument.

This commands will not return the correct instrument firmware version if an RSA is disconnected and then another connected without using the Search for Instruments tool in the SignalVu-PC application to find and connect to the new instrument.

SYSTem:SVPC:INSTrument:MODEL? (Query Only)

Available when an RSA is connected to SignalVu-PC. Queries the model of the connected instrument.

Group

System commands

Syntax

```
SYSTem:SVPC:INSTrument:MODEL?
```

Arguments

None.

Returns

Model of connected instrument.

This commands will not return the correct model if an RSA is disconnected and then another connected without using the Search for Instruments tool in the SignalVu-PC application to find and connect to the new instrument.

SYSTem:SVPC:INSTrument:OPTions? (Query Only)

Available when an RSA is connected to SignalVu-PC. Queries the options for the connected instrument.

Group

System commands

Syntax

`SYSTem:SVPC:INSTrument:OPTions?`

Arguments

None.

Returns

Model of connected instrument.

This commands will not return the correct options if an RSA is disconnected and then another connected without using the Search for Instruments tool in the SignalVu-PC application to find and connect to the new instrument.

SYSTem:SVPC:SOURce<x>:ADD

Adds Measurement Source.

Conditions

It should be connected to MSO/LPD/DPO7/SX to set.

Group

System commands

Syntax

`SYSTem:SVPC:SOURce<x>:ADD`

Related Commands

SYSTem:SVPC:SOURce<x>:REMOve (on page 2563)

SYSTem:SVPC:SOURce<x>:RENAme (on page 2564)

SYSTem:SVPC:SOURce<x>:REMOve:ALL (on page 2564)

Arguments

`<x>` specifies the measurement source index, 1 for Source1.

Returns

None

Examples

`SYSTem:SVPC:SOURce1:ADD` adds measurement source under source configuration.

SYSTem:SVPC:SOURce<x>:REMOve

Removes latest user defined measurement Source from source configuration.

Conditions

It should be connected to MSO/LPD/DPO7/SX to set.

Group

System commands

Syntax

SYSTem:SVPC:SOURce<x>:REMOve

Related Commands

SYSTem:SVPC:SOURce<x>:ADD (on page 2563)

SYSTem:SVPC:SOURce<x>:RENAme (on page 2564)

SYSTem:SVPC:SOURce<x>:REMOve:ALL (on page 2564)

Arguments

<x> specifies the measurement source index, 1 for Source1.

Returns

None

Examples

SYSTem:SVPC:SOURce1:REMOve removes the measurement source from source configuration.

SYSTem:SVPC:SOURce<x>:REMOve:ALL

Removes all user defined measurement sources from source configuration.

Conditions

It should be connected to MSO/LPD/DPO7/SX to set.

Group

System commands

Syntax

SYSTem:SVPC:SOURce?:REMOve:ALL

Related Commands

SYSTem:SVPC:SOURce<x>:ADD
SYSTem:SVPC:SOURce<x>:REMOve
SYSTem:SVPC:SOURce<x>:RENAme

Arguments

None

Returns

None

Examples

SYSTem:SVPC:SOURce:REMOve:ALL, removes all user defined measurement sources from source configuration.

SYSTem:SVPC:SOURce<x>:RENAme

Sets or queries measurement source name under source configuration.

Conditions

It should be connected to MSO/LPD/DPO7/SX to set or query.

Group

System commands

Syntax

```
SYSTem:SVPC:SOURce<x>:RENAme <source_name>
SYSTem:SVPC:SOURce<x>:RENAme?
```

Related Commands

SYSTem:SVPC:SOURce<x>:ADD (on page 2563)

SYSTem:SVPC:SOURce<x>:REMOve (on page 2563)

SYSTem:SVPC:SOURce<x>:REMOve:ALL (on page 2564)

Arguments

<x> specifies measurement source index, 1 for Source1.

<source_name> specifies custom source name.

Returns

Measurement Source name

Examples

SYSTem:SVPC:SOURce1:RENAme "Wifi" renames the measurement source "Source 1" to "Wifi".

SYSTem:SVPC:SOURce1:RENAme? returns the measurement Source "Source 1" name.

SYSTem:TIME

Sets or queries the time (hours, minutes, and seconds). This command is equivalent to the time setting through the Windows Control Panel.

Conditions

Measurement views: All

Group

System commands

Syntax

```
SYSTem:TIME <hour>,<minute>,<second>
SYSTem:TIME?
```

Related Commands

SYSTem:DATE (on page 2541)

Arguments

<hour>::<NRf> specifies the hours. Range: 0 to 23.

<minute>::<NRf> specifies the minutes. Range: 0 to 59.

<second>::<NRf> specifies the seconds. Range: 0 to 59. These values are rounded to the nearest integer.

*RST has no effect on the settings.

Examples

`SYSTEM:TIME 10,15,30` sets the time to 10:15:30.

SYSTem:VERSion? (Query Only)

Returns the SCPI version number for which the analyzer complies.

Conditions

Measurement views: All

Group

System commands

Syntax

```
SYSTem:VERSion?
```

Arguments

None

Returns

`<NR2>` has the form YYYY.V where the Ys represent the year-version (for example, 1999) and the V represents an approved revision number for that year.

Examples

`SYSTEM:VERSION?` might return `1999.0` for the SCPI version.

Trace Commands

TRACe:{AM|FM|PM}:DETection

Sets or queries the display detector, the method to be used for decimating traces to fit the available horizontal space on screen. The number of horizontal pixels on screen is generally smaller than that of waveform data points. When actually displayed, the waveform data is therefore thinned out, according to the number of pixels, for being compressed.

Conditions

Measurement views: AM, FM, PM

Group

Trace commands

Syntax

```
TRACe:{AM|FM|PM}:DETection { AVERage | POSitive | NEGative }
TRACe:{AM|FM|PM}:DETection?
```

Arguments

`AVERage` displays the average data value for each pixel.

`POSitive` displays the maximum data value for each pixel.

`NEGative` displays the minimum data value for each pixel.

Examples

`TRACE:AM:DETECTION AVERage` specifies that the trace displays the average data value for each pixel.

TRACe:{AM|FM|PM}:FREeze

Determines whether or not to freeze the trace display in the AM/FM/PM view.

Conditions

Measurement views: AM, FM, PM

Group

Trace commands

Syntax

```
TRACe:{AM|FM|PM}:FREeze { OFF | ON | 0 | 1 }
TRACe:{AM|FM|PM}:FREeze?
```

Arguments

`OFF` or 0 updates the trace display normally.

`ON` or 1 stops updating the trace display.

Examples

`TRACE:AM:FREEZE ON` freezes the trace display.

TRACe:{AM|FM|PM}:FUNCtion

Sets or queries the trace function in the AM/FM/PM view.

Conditions

Measurement views: AM, FM, PM

Group

Trace commands

Syntax

```
TRACe:{AM|FM|PM}:FUNCtion { NORMal }
TRACe:{AM|FM|PM}:FUNCtion?
```

Arguments

`NORMal` selects the normal display.

Examples

`TRACE:AM:FUNCTION NORMal` selects the normal display in the AM view.

TRACe:DIQVtime[:MEASview<y>]:ENABle:I

Enables or disables the I trace in the Demod I&Q versus Time measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Demod I&Q versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod I&Q versus Time).

Group

Trace commands

Syntax

```
TRACe:DIQVtime[:MEASview<y>]:ENABle:I { OFF | ON | 0 | 1 }
TRACe:DIQVtime[:MEASview<y>]:ENABle:I?
```

Arguments

`OFF` or 0 hides the I trace.

`ON` or 1 shows the I trace.

Examples

`TRACe:DIQVtime:MEASview1:ENABle:I ON` shows the I trace in the Demod I&Q versus Time measurement view 1.

TRACe:DIQVtime[:MEASview<y>]:ENABle:Q

Enables or disables the Q trace in the Demod I&Q versus Time measurement.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Demod I&Q versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod I&Q versus Time).

Group

Trace commands

Syntax

```
TRACe:DIQVtime[:MEASview<y>]:ENABle:Q { OFF | ON | 0 | 1 }
TRACe:DIQVtime[:MEASview<y>]:ENABle:Q?
```

Arguments

`OFF` or 0 hides the Q trace.

`ON` or 1 shows the Q trace.

Examples

`TRACe:DIQVtime:MEASview1:ENABle:Q ON` shows the Q trace in the Demod I&Q versus Time Measurement view 1.

TRACe:DIQVtime[:MEASview<y>]:MODE

Sets or queries whether to display the Demod I&Q versus Time trace as vectors or symbols (points).

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement view: Demod I&Q versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod I&Q versus Time).

Group

Trace commands

Syntax

```
TRACe:DIQVtime[:MEASview<y>]:MODE { VECTors | SYMBols }
```

Arguments

`VECTors` displays the trace in a format that uses lines to connecting points in the trace display.

`SYMBols` displays the trace in a format that does not use lines to connect points in the trace display.

Examples

`TRACe:DIQVtime:MEASview1:MODE SYMBols` sets the display to show symbols without lines connecting points Measurement view 1.

TRACe:DIQVtime[:MEASview<y>]:SELect:I

Selects the I trace in the Demod I&Q versus Time. The query version of this command returns whether the I trace is selected or not.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Demod I&Q versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod I&Q versus Time).

Group

Trace commands

Syntax

```
TRACe:DIQVtime[:MEASview<y>]:SELect:I
TRACe:DIQVtime[:MEASview<y>]:SELect:I?
```

Arguments

None

Returns

0 indicates that the I trace is deselected.

1 indicates that the I trace is selected.

Examples

`TRACe:DIQVtime:MEASview1:SELect:I` selects the I trace in the Demod I&Q versus Time Measurement view 1.

TRACe:DIQVtime[:MEASview<y>]:SELect:Q

Selects the Q trace in the Demod I&Q versus Time. The query version of this command returns whether the Q trace is selected or not.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Demod I&Q versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod I&Q versus Time).

Group

Trace commands

Syntax

```
TRACe:DIQVtime[:MEASview<y>]:SELect:Q
TRACe:DIQVtime[:MEASview<y>]:SELect:Q?
```

Arguments

None

Returns

0 indicates that the Q trace is deselected.

1 indicates that the Q trace is selected.

Examples

`TRACe:DIQVtime[:MEASview1]:SELect:Q` selects the Q trace in the Demod I&Q versus Time Measurement view 1.

TRACe:EDIagram[:MEASview<y>]:ENABle:I

Determines whether to show or hide the I trace in the Eye diagram.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Eye diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Eye diagram).

Group

Trace commands

Syntax

```
TRACe:EDIagram[:MEASview<y>]:ENABle:I { OFF | ON | 0 | 1 }
TRACe:EDIagram[:MEASview<y>]:ENABle:I?
```

Arguments

`OFF` or `0` hides the I trace.

`ON` or `1` shows the I trace.

Examples

`TRACe:EDIagram:MEASview2:ENABle:I ON` shows the I trace in the eye diagram Measview 2.

TRACe:EDIagram[:MEASview<y>]:ENABle:Q

Determines whether to show or hide the Q trace in the Eye diagram.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Eye diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Eye diagram).

Group

Trace commands

Syntax

```
TRACe:EDIagram[:MEASview<y>]:ENABle:Q { OFF | ON | 0 | 1 }
TRACe:EDIagram[:MEASview<y>]:ENABle:Q?
```

Arguments

`OFF` or `0` hides the Q trace.

`ON` or `1` shows the Q trace.

Examples

`TRACe:EDIagram:MEASview2:ENABle:Q ON` shows the Q trace in the eye diagram Measview 2.

TRACe:EDIagram[:MEASview<y>]:SELect:I

Selects the I trace in the Eye diagram. When used as query, it returns whether the I trace is selected or not.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Eye diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Eye diagram).

Group

Trace commands

Syntax

```
TRACe:EDIagram[:MEASview<y>]:SELect:I
TRACe:EDIagram[:MEASview<y>]:SELect:I?
```

Arguments

None

Returns

0 indicates that the I trace is deselected.

1 indicates that the I trace is selected.

Examples

TRACe:EDIagram:MEASview2:SELect:I selects the I trace in the eye diagram for measview2.

TRACe:EDIagram[:MEASview<y>]:SELect:Q

Selects the Q trace in the eye diagram. The query version of this command returns whether the Q trace is selected or not.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Eye diagram

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: Eye diagram).

Group

Trace commands

Syntax

```
TRACe:EDIagram[:MEASview<y>]:SELect:Q
TRACe:EDIagram[:MEASview<y>]:SELect:Q?
```

Arguments

None

Returns

0 indicates that the Q trace is deselected.

1 indicates that the Q trace is selected.

Examples

TRACe:EDIagram:MEASview2:SELect:Q selects the Q trace in the eye diagram Measview 2.

TRACe:EMC:AMBIent:COUNt

Sets or queries the count value of specified ambient trace in EMC view.

<value> = <NR1> count value

Conditions

Measurement view: EMC - EMI Display

Group

Trace commands

Syntax

```
TRACe:EMC:AMBIent:COUNt <value>
TRACe:EMC:AMBIent:COUNt?
```

Arguments

<value> = <NR1>

Returns

<NR1>

Examples

TRACE:EMC:AMBIENT:COUNT might return 20, indicating that the count value of ambient trace 2 is 20.

TRACe:EMC:AMBIent:COUNt:ENABle

Sets or queries to enable or disable the count check box for specified ambient trace in EMC view.

Conditions

Measurement view: EMC - EMI Display

Group

Trace commands

Syntax

```
TRACe:EMC:AMBIent:COUNt:ENABle {0 | OFF | 1 | ON}
TRACe:EMC:AMBIent:COUNt:ENABle?
```

Arguments

0 or OFF disables the count check box 1 or ON enables the count check box

Returns

<Boolean> 0, 1

Examples

TRACE:EMC:AMBIENT:COUNT:ENABLE might return 1, indicating that count check box is enabled.

TRACe:EMC:AMBIent:COUNt:RESet (No Query Form)

Resets the count value of live ambient trace in EMC view.

Conditions

Measurement view: EMC - EMI Display

Group

Trace commands

Syntax

```
TRACe:EMC:AMBIent:COUNt:RESet
```

Arguments

None

Returns

None

Examples

`TRACE:EMC:AMBIENT:COUNT:RESET` resets the count value of live ambient trace

TRACe:EMC:AMBIent:ENABle

Sets or queries to show or hide the ambient trace on plot by enabling or disabling the show check box in EMC view.

Conditions

Measurement view: EMC - EMI Display

Group

Trace commands

Syntax

```
TRACe:EMC:AMBIent:ENABle {0 | OFF | 1 | ON}
TRACe:EMC:AMBIent:ENABle?
```

Arguments

`0` or `OFF` disables the show check box and hides the ambient trace 1 or ON enables the show check box and displays the ambient trace

Returns

`<Boolean>` 0, 1

Examples

`TRACE:EMC:AMBIENT:ENABLE?` might return `0`, indicating that the selected ambient trace is hidden.

TRACe:EMC:AMBIent:FREeze

Sets or queries to freeze the ambient trace on plot by enabling or disabling the freeze check box in EMC view.

Conditions

Measurement view: EMC - EMI Display

Group

Trace commands

Syntax

```
TRACe:EMC:AMBIent:FREeze {0 | OFF | 1 | ON}
TRACe:EMC:AMBIent:FREeze?
```

Arguments

`0` or `OFF` disables the freeze check box 1 or ON enables the freeze check box

Returns

`<Boolean>` 0, 1

Examples

TRACE:EMC:AMBIENT:FREEZE? might return 1, indicating that the selected ambient trace is frozen.

TRACe:EMC:AMBIent:FUNCtion

Sets or queries the scan function for the selected ambient trace in EMC View.

Conditions

Measurement view: EMC - EMI Display

Group

Trace commands

Syntax

```
TRACe:EMC:AMBIent:FUNCtion {NONE | MAXH | AVER | AVGL}
TRACe:EMC:AMBIent:FUNCtion?
```

Arguments

NONE = Normal is the function selected

MAXH = Max Hold is the function selected

AVER = Avg(VRMS) is the function selected

AVGL = Avg (of logs) is the function selected

Returns

<String> See Arguments

Examples

TRACE:EMC:AMBIENT:FUNCTION might return AVER, indicating the scan function selected for ambient trace is Avg(VRMS).

TRACe:EMC:AMBIent:SELect

Sets or queries the selection of ambient trace in live trace of EMC View.

Conditions

Measurement view: EMC - EMI Display

Group

Trace commands

Syntax

```
TRACe:EMC:AMBIent:SELect
TRACe:EMC:AMBIent:SELect?
```

Arguments

None

Returns

<Boolean> 0, 1

Examples

TRACE:EMC:AMBIENT:SELECT might return 1, indicating that ambient trace 1 is selected in live trace.

TRACe:EMC:MATH:ENABle

Sets or queries to show or hide the math trace on plot by enabling or disabling the show check box in EMC view.

Conditions

Measurement view: EMC - EMI Display

Group

Trace commands

Syntax

```
TRACe:EMC:MATH:ENABle {0 | OFF | 1 | ON}
TRACe:EMC:MATH:ENABle?
```

Arguments

0 or OFF disables the show check box and hides the math trace 1 or ON enables the show check box and displays the math trace

Returns

<Boolean> 0, 1

Examples

TRACE:EMC:MATH:ENABLE? might return 1, indicating that the math trace is enabled (displayed).

TRACe:EMC:MATH:FREeze

Sets or queries to freeze the math trace on plot by enabling or disabling the freeze check box in EMC view.

Conditions

Measurement view: EMC - EMI Display

Group

Trace commands

Syntax

```
TRACe:EMC:MATH:FREeze {0 | OFF | 1 | ON}
TRACe:EMC:MATH:FREeze?
```

Arguments

0 or OFF disables the freeze check box 1 or ON enables the freeze check box

Returns

<Boolean> 0, 1

Examples

TRACE:EMC:MATH:FREEZE? might return 1, indicating that selected math trace is frozen.

TRACe:EMC:MATH:LEFToperand

Sets or queries the trace for left operand to obtain a math trace in EMC view.

Conditions

Measurement view: EMC - EMI Display

Group

Trace commands

Syntax

```
TRACe:EMC:MATH:LEFToperand <arg>
TRACe:EMC:MATH:LEFToperand?
```

Arguments

```
<arg> = {scan1 | scan2 | ambient | recall1 | recall2 | recall3 | recall4 | recall5}
```

Returns

<String>

Examples

`TRACE:EMC:MATH:LEFTOPERAND` selects ambient trace as left operand.

TRACe:EMC:MATH:RIGHtoperand

Sets or queries the trace for right operand to obtain a math trace in EMC view.

Conditions

Measurement view: EMC - EMI Display

Group

Trace commands

Syntax

```
TRACe:EMC:MATH:RIGHtoperand <arg>
TRACe:EMC:MATH:RIGHtoperand?
```

Arguments

```
<arg> = {scan1 | scan2 | ambient | recall1 | recall2 | recall3 | recall4 | recall5}
```

Returns

```
<String>
```

Examples

`TRACE:EMC:MATH:RIGHTOPERAND?` might return `scan1`, indicating that the right operand selected is scan1 trace.

TRACe:EMC:MATH:SELect

Sets or queries the selection of math trace in live trace of EMC View.

Conditions

Measurement view: EMC - EMI Display

Group

Trace commands

Syntax

```
TRACe:EMC:MATH:SELect
TRACe:EMC:MATH:SELect?
```

Arguments

None

Returns

<Boolean> 0, 1

Examples

TRACE:EMC:MATH:SELECT might return 1, indicating that math trace is selected.

TRACe:EVM[:MEASview<y>]:MODE

Sets or queries whether to display the EVM versus Time trace as vectors or symbols (points).

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement view: EVM versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: EVM versus Time).

Group

Trace commands

Syntax

```
TRACe:EVM[:MEASview<y>]:MODE { VECTors | SYMBols }
TRACe:EVM[:MEASVIEW<y>]:MODE?
```

Arguments

VECTors displays the trace in a format that uses lines to connecting points in the trace display.

SYMBols displays the trace in a format that does not use lines to connect points in the trace display.

Examples

TRACe:EVM:MEASVIEW2:MODE VECTors sets the display to connect points in the display for measview 2 of EVM versus Time.

TRACe:FDVTime[:MEASview<y>]:MODE

Sets or queries whether to display the Frequency Deviation versus Time trace as vectors or symbols (points).

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement view: Frequency deviation versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Frequency deviation versus Time).

Group

Trace commands

Syntax

```
TRACe:FDVTime[:MEASview<y>]::MODE { VECTors | SYMBols }
TRACe:FDVTime[:MEASView<y>]:MODE?
```

Arguments

`VECTors` display the trace in a format that uses lines to connect points in the trace display.

`SYMBols` display the trace in a format that does not use lines to connect points in the trace display.

Examples

`TRACE:FDVTime:MEASview2:MODE VECTORS` sets the display to connect points in the display for measview 2 of Frequency deviation versus Time.

TRACe:{FSETtling|PSETtling}:AVERage:COUNt

Sets or queries the number of traces to combine. This command is effective when you select AVERage with the **s** command.

Conditions

Measurement views: Frequency and Phase Settling Time

Group

Trace commands

Syntax

```
TRACe:{FSETtling|PSETtling}:AVERage:COUNt <number>
TRACe:{FSETtling|PSETtling}:AVERage:COUNt?
```

Related Commands

TRACe:{FSETtling|PSETtling}:AVERage:ENABle (on page 2580)

Arguments

`<number>::<NR1>` specifies the number of traces to combine for averaging.

Range: 1 to 10000.

Examples

`TRACE1:PSETTLING:AVERAGE:COUNT 64` sets the average count to 64 for Trace 1.

TRACe:{FSETtling|PSETtling}:AVERage:COUNt:CURRent? (Query Only)

Queries the current running average count value. The returned value is valid (non-zero) only while the instrument is acquiring and averaging a signal. Use the `TRACe:{FSETtling|PSETtling}:AVERage:COUNt` command to determine the setting for Average.

Conditions

Measurement views: Frequency and Phase Settling Time

Group

Trace commands

Syntax

```
TRACe:{FSETtling|PSETtling}:AVERage:COUNt:CURRent?
```

Related Commands

TRACe:{FSETtling|PSETtling}:AVERage:COUNt (on page 2579)

Examples

`TRACE1:PSETTLING:AVERAGE:COUNT?` might return the value 64.

TRACe:{FSETtling|PSETtling}:AVERage:ENABle

Determines whether to enable or disable averaging the trace(s).

Conditions

Measurement views: Frequency and Phase Settling Time

Group

Trace commands

Syntax

```
TRACe:{FSETtling|PSETtling}:AVERage:ENABle { OFF | ON | 0 | 1 }
TRACe:{FSETtling|PSETtling}:AVERage:ENABle?
```

Arguments

`OFF` or 0 disables averaging.

`ON` or 1 enables averaging.

Examples

`TRACE:PSETTLING:AVERAGE:ENABLE ON` enables the Phase Settling Time averaging.

TRACe:{FSETtling|PSETtling}:RESet (No Query Form)

If trace averaging is enabled, this command resets the current averaged trace to 0.

Conditions

Measurement views: Frequency and Phase Settling Time

Group

Trace commands

Syntax

```
TRACe:{FSETtling|PSETtling}:RESet
```

Examples

`TRACE1:PSETTLING:RESET` resets the current averaged trace to 0.

TRACe:{FSETtling|PSETtling}:SMOothing:COUNt

Sets or queries the number of data points to take the moving average for smoothing the traces. This command applies to both Trace 1 and Trace 2.

Conditions

Measurement views: Frequency and Phase Settling Time

Group

Trace commands

Syntax

```
TRACe:{FSETtling|PSETtling}:SMOothing:COUNt <number>
TRACe:{FSETtling|PSETtling}:SMOothing:COUNt?
```

Arguments

`<number>::<NR1>` specifies the number of data points to take the moving average for smoothing. Range: 2 to 1000.

Examples

`TRACE:FSETTLING:SMOOTHING:COUNT 16` sets the Frequency Settling Time smoothing count to 16.

TRACe:{FSETtling|PSETtling}:SMOothing:ENABle

Determines whether to enable or disable smoothing the trace(s).

Conditions

Measurement views: Frequency and Phase Settling Time

Group

Trace commands

Syntax

```
TRACe:{FSETtling|PSETtling}:SMOothing:ENABle { OFF | ON | 0 | 1 }
TRACe:{FSETtling|PSETtling}:SMOothing:ENABle?
```

Arguments

`OFF` or 0 disables smoothing.

`ON` or 1 enables smoothing.

Examples

`TRACE:PSETTLING:SMOOTHING:ENABLE ON` enables the Phase Settling Time smoothing.

TRACe<x>:FVTime[:MEASview<y>]

Determines whether or not to show the trace in the Frequency versus Time view.

Conditions

Measurement views: Frequency versus Time

Group

Trace commands

Syntax

```
TRACe<x>:FVTime[:MEASview<y>] { OFF | ON | 0 | 1 }
TRACe<x>:FVTime[:MEASview<y>]?
```

Arguments

<x> = 1 to 7; Trace 2, Trace 3 and Trace 4 (math trace) are invalid.

- Trace5: Bitmap Trace
- Trace6: Ogram
- Trace7: Ogram Line

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value of y is 1.

OFF or 0 hides the trace in the Frequency versus Time view.

ON or 1 shows the trace in the Frequency versus Time view.

Examples

TRACE1:FVTIME:MEASview1 ON shows the trace in the Frequency versus Time view.

TRACe<x>:FVTime[:MEASview<y>]:AVERage:COUNt

Sets or queries the number of traces to combine. This command is effective when you select AVERage with the TRACe:FVTime[:MEASview<y>]:FUNCtion command.

Conditions

Measurement views: Frequency versus Time

Group

Trace commands

Syntax

```
TRACe<x>:FVTime[:MEASview<y>]:AVERage:COUNt <number>
TRACe<x>:FVTime[:MEASview<y>]:AVERage:COUNt?
```

Arguments

<x> = 1 to 3; Trace 4 (math trace) and Trace 5 (spectrogram) are invalid.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<number>::<NR1> specifies the number of traces to combine for averaging.

Range: 1 to 10000.

Examples

`TRACE1:FVTIME:MEASview1:AVERAGE:COUNT 64` sets the average count to 64.

TRACe<x>:FVTime[:MEASview<y>]:COLor:CURVe

Sets or queries how colors are mapped to the signal density in the Frequency versus Time bitmap display. The mapping can be linear (Curve = 1), or it can be set to concentrate the resolution on the lower level of the range (Curve > 1) or the mapping can be set to show the best resolution on the upper range of density or hit count (Curve = 0.1 - 0.99). This command is only valid for the Bitmap display.

The parameter <x> = 5; only Trace5 is valid.

Conditions

Measurement views: Frequency versus Time

Group

Trace commands

Syntax

`TRACe<x>:FVTime[:MEASview<y>]:COLor:CURVe <value>`

`TRACe<x>:FVTime[:MEASview<y>]:COLor:CURVe?`

Arguments

`<value>` = <NRf> specifies how colors are mapped to the signal density. Range: 0.1 to 10 (the default value is 2).

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. The default value of y is 1.

Examples

`TRACe5:FVTIME:COLor:CURVe 1.5` concentrates the resolution on the lower level of the range on the Bitmap display.

TRACe<x>:FVTime[:MEASview<y>]:COLor:SCALe:AUTO (No Query Form)

Automatically adjusts the Max and Min color settings to display the broadest range of colors in the Frequency versus Time bitmap display. This command is only valid for the Bitmap display.

The parameter <x> = 5; only Trace5 is valid.

Conditions

Measurement views: Frequency versus Time

Group

Trace commands

Syntax

`TRACe<x>:FVTime[:MEASview<y>]:COLor:SCALe:AUTO`

Arguments

`<x>` = 5 specifies the auto-color setting for the Bitmap trace.

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. The default value of y is 1.

Examples

`TRACe5:FVTIME:COLor:SCALe:AUTO` autosets the Bitmap display color scheme.

TRACe<x>:FVTime[:MEASview<y>]:COUNt

Sets or queries how many acquisitions run in the single acquisition mode for the Max or Min Hold trace in the Frequency versus Time measurement. This command is effective when `TRACe<x>:FVTime[:MEASview<y>]:FUNCtion` is set to `MAXHold` or `MINHold` and `INITiate:CONTinuous` is set to OFF.

Conditions

Measurement views: Frequency versus Time

Group

Trace commands

Syntax

```
TRACe<x>:FVTime[:MEASview<y>]:COUNt <number>
TRACe<x>:FVTime[:MEASview<y>]:COUNt?
```

Arguments

<x> = 1 to 3; Trace 4 (math trace) and Trace 5 (spectrogram) are invalid.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

`<number>::<NR1>` specifies the count for Max/Min Hold.

Range: 1 to 10000.

Examples

`TRACe1:FVTime:MEASview1:COUNt 32` sets the count to 32 for the Max/Min Hold trace.

TRACe<x>:FVTime[:MEASview<y>]:COUNt:ENABle

Determines whether to enable or disable the count for the Max or Min Hold trace in the Frequency versus Time view. This command is effective when `TRACe<x>:FVTime[:MEASview<y>]:FUNCtion` is set to `MAXHold` or `MINHold`.

Conditions

Measurement views: Frequency versus Time

Group

Trace commands

Syntax

```
TRACe<x>:FVTime[:MEASview<y>]:COUNt:ENABle { OFF | ON | 0 | 1 }
TRACe<x>:FVTime[:MEASview<y>]:COUNt:ENABle?
```

Arguments

`<x>` = 1 to 3; Trace 4 (math trace) and Trace 5 (spectrogram) are invalid.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

OFF or 0 disables the count for the Max/Min Hold trace.

ON or 1 enables the count for the Max/Min Hold trace.

Examples

TRACe1:FVTime:MEASview1:COUNt:ENABle ON enables the Max/Min Hold count.

TRACe<x>:FVTime[:MEASview<y>]:COUNt:RESet (No Query Form)

Clears the Max or Min Hold data and counter, and restarts the process in the Frequency versus Time view. This command is effective when TRACe<x>:FVTime[:MEASview<y>]:FUNCtion is set to MAXHold or MINHold.

Conditions

Measurement views: Frequency versus Time

Group

Trace commands

Syntax

```
TRACe<x>:FVTime[:MEASview<y>]:COUNt:RESet
```

Arguments

<x> = 1 to 3; Trace 4 (math trace) and Trace 5 (spectrogram) are invalid.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Examples

TRACe1:FVTime:MEASview1:COUNt:RESet clears the Max/Min Hold data and counter, and restarts the process.

TRACe<x>:FVTime[:MEASview<y>]:DETection

Enables or queries the type of detection for the specified trace. The parameter <x> represents traces 1 to 4.

Conditions

Measurement views: Frequency versus Time

Group

Trace commands

Syntax

```
TRACe<x>:FVTime[:MEASview<y>]:DETection {AVERage |POSitive | NEGative | POSNegative | SAMPle}
TRACe<x>:FVTime[:MEASview<y>]:DETection?
```

Arguments

<x> = 1 to 6.

- Trace 4: Math Trace (not applicable)
- Trace 5: Bitmap Trace (not applicable)
- Trace 6: Ogram

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

`AVERage` displays the average data value for each pixel.

`POSitive` displays the maximum data value for each pixel.

`NEGative` displays the minimum data value for each pixel.

`POSNegative` displays the maximum and minimum data values for each pixel.

`SAMPle` displays the first sample value received for each pixel.

Examples

`TRACe2:FVTime:MEASview1:DETection POSitive` enables positive detection on Trace2.

TRACe<x>:FVTime[:MEASview<y>]:DOT:PERSistent

Determines whether to enable or disable the dot persistence for the bitmap trace (Trace 5) in the Frequency versus Time view.

The parameter <x> = 5; Only Trace 5 (bitmap trace) is valid.

Conditions

Measurement views: Frequency versus Time

Group

Trace commands

Syntax

`TRACe<x>:FVTime[:MEASview<y>]:DOT:PERSistent { OFF | ON | 0 | 1 }`

`TRACe<x>:FVTime[:MEASview<y>]:DOT:PERSistent?`

Arguments

`OFF` or `0` disables the dot persistence.

`ON` or `1` enables the dot persistence.

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`TRACE5:FVTIME:DOT:PERSISTENT ON` enables the dot persistence in the Frequency versus Time view.

TRACe<x>:FVTime[:MEASview<y>]:DOT:PERSistent:TYPE

Sets or queries the persistence type for the bitmap trace (Trace 5) in the Frequency versus Time view.

The parameter <x> = 5; Only Trace 5 (bitmap trace) is valid.

Conditions

Measurement views: Frequency versus Time

Group

Trace commands

Syntax

`TRACe<x>:FVTime[:MEASview<y>]:DOT:PERSistent:TYPE { VARiable | INFinite }`

`TRACe<x>:FVTime[:MEASview<y>]:DOT:PERSistent:TYPE?`

Arguments

`VARiable`: Selects the variable persistence display which leaves acquired data points on the display for a period of time specified by the `TRACe<x>:FVTime[:MEASview<y>]:DOT:PERSistent:VARiable` command.

`INFinite`: Selects the infinite persistence display which accumulates data points on the display indefinitely.

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`TRACE5:FVTIME:DOT:PERSISTENT:TYPE VARiable` selects the variable persistence display.

TRACe<x>:FVTime[:MEASview<y>]:DOT:PERSistent:VARiable

Sets or queries how long data points are displayed. This command is effective when `TRACe<x>:FVTime[:MEASview<y>]:DOT:PERSistent:TYPE` is set to `VARiable`. This affects the display only.

The parameter <x> = 5; Only Trace 5 (bitmap trace) is valid.

Conditions

Measurement views: Frequency versus Time

Group

Trace commands

Syntax

`TRACe<x>:FVTime[:MEASview<y>]:DOT:PERSistent:VARiable <number>`

`TRACe<x>:FVTime[:MEASview<y>]:DOT:PERSistent:VARiable?`

Arguments

`<number>` = <NRf> specifies the how long waveform points are displayed on the screen. Range: 0 s to 1000 s (the default value is 10 ms).

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`TRACE5:FVTIME:DOT:PERSISTENT:VARIABLE 20` specifies that the waveform points are displayed on the screen for a period of 20 before they disappear.

TRACe<x>:FVTime[:MEASview<y>]:FREeze

Determines whether or not to freeze the display of the trace in the Frequency versus Time measurement.

Conditions

Measurement views: Frequency versus Time

Group

Trace commands

Syntax

```
TRACe<x>:FVTime[:MEASview<y>]:FREeze { OFF | ON | 0 | 1 }
TRACe<x>:FVTime[:MEASview<y>]:FREeze?
```

Arguments

<x> = 1 to 7.

- Trace 4: Math trace
- Trace 5: Bitmap trace
- Trace 6: Ogram
- Trace 7: Ogram Line

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

OFF or 0 updates the display of the trace normally.

ON or 1 stops updating the display of the trace.

Examples

TRACE1:FVTIME:MEASview1:FREEZE ON stops updating the display of the trace.

TRACe<x>:FVTime[:MEASview<y>]:FUNCtion

Sets or queries the trace function in the Frequency versus Time measurement.

Conditions

Measurement views: Frequency versus Time

Group

Trace commands

Syntax

```
TRACe<x>:FVTime[:MEASview<y>]:FUNCtion { NORMal | AVERage | MAXHold | MINHold}
TRACe<x>:FVTime[:MEASview<y>]:FUNCtion?
```

Arguments

<x> = 1 to 3; Trace 4 (math trace) and Trace 5 (spectrogram) are invalid.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

NORMal selects the normal waveform display.

AVERage selects the Average display that indicates the average frequency drift at each time point.

MAXHold selects the Max Hold display that indicates the maximum frequency drift at each time point.

MINHold selects the Min Hold display that indicates the minimum frequency drift at each time point.

Examples

TRACE1:FVTIME:MEASview1:FUNCTIONMAX Hold displays the Max Hold trace in the Frequency versus Time measurement.

TRACe<x>:FVTime[:MEASview<y>]:SELect

Sets or queries the selected trace.

Conditions

Applicable measurement view must be active.

Group

Trace commands

Syntax

`TRACe<x>:FVTime[:MEASview<y>]:SELect {ON | OFF | 0 | 1}`

`TRACe<x>:FVTime[:MEASview<y>]:SELect?`

Arguments

`ON` or `1` selects the specified trace

The parameter <x> = 1 to 7.

- Trace5: Bitmap Trace
- Trace6: Ogram
- Trace7: Ogram Line

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Returns

`0` indicates the specified trace is not the selected trace.

1 indicates the specified trace is the selected trace.

Examples

`TRACe6:FVTime:MEASview1:SELect 1` selects Trace6 as the selected trace for measview 1 of Frequency vs time.

TRACe<x>:FVTime[:MEASview<y>]:SELect:LINe

Sets or queries the line number to send to the Frequency vs time ogram display.

Conditions

Applicable measurement view must be active.

Group

Trace commands

Syntax

`TRACe<x>:FVTime[:MEASview<y>]:SELect:LINe <number>`

`TRACe<x>:FVTime[:MEASview<y>]:SELect:LINe?`

Related Commands

TRACe<x>:FVTime[:MEASview<y>] (on page 2582)

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional. By default y is 1.

<x> = 6 for ogram trace.

`<number>::=<NR1>` specifies the line number to send to the display.

Range: 0 to the maximum line number of the ogram displayed on screen.

Returns

`<NRf>` specifies the line number to send to the spectrum display.

Examples

`TRACe6:FVTime:MEASview1:SELect:LINe 75` selects Line #75 in the ogram to send to the time plot display.

TRACe<x>:IQVTime[:MEASview<y>]:AVERage:COUNt

Sets or queries the number of traces to combine. This command works for both I and Q traces when you select AVERage in the `TRACe<x>:IQVTime[:MEASview<y>]:FUNCtion` command.

Conditions

Measurement views: RF I & Q versus Time

Group

Trace commands

Syntax

```
TRACe<x>:IQVTime[:MEASview<y>]:AVERage:COUNt <number>
TRACe<x>:IQVTime[:MEASview<y>]:AVERage:COUNt?
```

Arguments

`<x>` = 1 to 3; Trace 4 (math trace) and Trace 5 (spectrogram) are invalid.

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

`<number>::<NR1>` specifies the number of traces to combine for averaging.

Range: 1 to 10000.

Examples

`TRACE1:IQVTIME:MEASview1:AVERAGE:COUNT 64` sets the average count to 64.

TRACe<x>:IQVTime[:MEASview<y>]:COUNt

Sets or queries how many acquisitions run in the single acquisition mode for the Max or Min Hold trace in the RF I & Q versus Time measurement. This command is effective when `TRACe<x>:IQVTime[:MEASview<y>]:FUNCtion` is set to `MAXHold` or `MINHold` and `INITiate:CONTinuous` is set to OFF.

Conditions

Measurement views: RF I & Q versus Time

Group

Trace commands

Syntax

```
TRACe<x>:IQVTime[:MEASview<y>]:COUNt <number>
TRACe<x>:IQVTime[:MEASview<y>]:COUNt?
```

Arguments

<x> = 1 to 3; Trace 4 (math trace) and Trace 5 (spectrogram) are invalid.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<number>::<NR1> specifies the count for Max/Min Hold.

Range: 1 to 10000.

Examples

TRACE1:IQVTIME:MEASview1:COUNT 32 sets the count to 32 for the Max/Min Hold trace.

TRACe<x>:IQVTime[:MEASview<y>]:COUNt:ENABle

Determines whether to enable or disable the count for the Max or Min Hold trace in the RF I & Q versus Time view. This command is effective when TRACe<x>:IQVTime[:MEASview<y>]:FUNCtion is set to MAXHold or MINHold.

Conditions

Measurement views: RF I & Q versus Time

Group

Trace commands

Syntax

```
TRACe<x>:IQVTime[:MEASview<y>]:COUNt:ENABle { OFF | ON | 0 | 1 }
TRACe<x>:IQVTime[:MEASview<y>]:COUNt:ENABle?
```

Arguments

<x> = 1 to 3; Trace 4 (math trace) and Trace 5 (spectrogram) are invalid.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

OFF or 0 disables the count for the Max/Min Hold trace.

ON or 1 enables the count for the Max/Min Hold trace.

Examples

TRACE1:IQVTIME:MEASview1:COUNT:ENABLE ON enables the Max/Min Hold count.

TRACe<x>:IQVTime[:MEASview<y>]:COUNt:RESet (No Query Form)

Clears the Max or Min Hold data and counter, and restarts the process in the RF I & Q versus Time view. This command is effective when TRACe<x>:IQVTime[:MEASview<y>]:FUNCtion is set to MAXHold or MINHold.

Conditions

Measurement views: RF I & Q versus Time

Group

Trace commands

Syntax

```
TRACe<x>:IQVTime[:MEASview<y>]:COUNt:RESet
```

Arguments

<x> = 1 to 3; Trace 4 (math trace) and Trace 5 (spectrogram) are invalid.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Examples

`TRACE1:IQVTIME:MEASview1:COUNT:RESET` clears the Max/Min Hold data and counter, and restarts the process.

TRACe<x>:IQVTime[:MEASview<y>]:DETection

Sets or queries the detection method for the RF I & Q versus Time view.

Conditions

Measurement views: RF I & Q versus Time

Group

Trace commands

Syntax

```
TRACe<x>:IQVTime[:MEASview<y>]:DETection { AVERage | POSitive | NEGative | POSNegative | SAMPle }
TRACe<x>:IQVTime[:MEASview<y>]:DETection?
```

Arguments

<x> = 1 to 3; Trace 4 (math trace) and Trace 5 (spectrogram) are invalid.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

`AVERage` displays the average data value for each pixel.

`POSitive` displays the maximum data value for each pixel.

`NEGative` displays the minimum data value for each pixel.

`POSNegative` displays the maximum and minimum data values for each pixel.

`SAMPle` displays the first sample value received for each pixel.

Examples

`TRACE1:IQVTIME:MEASview1:DETECTION AVERAGE` enables display of the average value for each pixel.

TRACe<x>:IQVTime[:MEASview<y>]:ENABle:I

Determines whether to show or hide the I trace in the RF I & Q versus Time measurement.

Conditions

Measurement views: RF I & Q versus Time

Group

Trace commands

Syntax

```
TRACe<x>:IQVTime[:MEASview<y>]:ENABle:I { OFF | ON | 0 | 1 }
TRACe<x>:IQVTime[:MEASview<y>]:ENABle:I?
```

Arguments

<x> = 1 to 3; Trace 4 (math trace) and Trace 5 (spectrogram) are invalid.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

OFF or 0 hides the I trace.

ON or 1 shows the I trace.

Examples

TRACE1:IQVTIME:MEASview1:ENABLE:I ON shows the I trace in the RF I & Q versus Time measurement.

TRACe<x>:IQVTime[:MEASview<y>]:ENABle:Q

Determines whether to show or hide the Q trace in the RF I & Q versus Time measurement.

Conditions

Measurement views: RF I & Q versus Time

Group

Trace commands

Syntax

```
TRACe<x>:IQVTime[:MEASview<y>]:ENABle:Q { OFF | ON | 0 | 1 }
TRACe<x>:IQVTime[:MEASview<y>]:ENABle:Q?
```

Arguments

<x> = 1 to 3; Trace 4 (math trace) and Trace 5 (spectrogram) are invalid.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

OFF or 0 hides the Q trace.

ON or 1 shows the Q trace.

Examples

TRACE1:IQVTIME:MEASview1:ENABLE:Q ON shows the Q trace in the IQ level versus Time measurement.

TRACe<x>:IQVTime[:MEASview<y>]:FREeze

Determines whether to freeze the IQ traces in the RF I & Q versus Time measurement.

Conditions

Measurement views: RF I & Q versus Time

Group

Trace commands

Syntax

```
TRACe<x>:IQVTime[:MEASview<y>]:FREeze { OFF | ON | 0 | 1 }
TRACe<x>:IQVTime[:MEASview<y>]:FREeze?
```

Arguments

<x> = 1 to 3; Trace 4 (math trace) and Trace 5 (spectrogram) are invalid.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

OFF or 0 updates IQ trace display normally.

ON or 1 stops updating IQ trace display.

Examples

TRACE1:IQVTIME:MEASview1:FREEZE ON freezes the IQ traces.

TRACe<x>:IQVTime[:MEASview<y>]:FUNCtion

Sets or queries the trace function in the RF I & Q versus Time measurement.

Conditions

Measurement views: RF I & Q versus Time

Group

Trace commands

Syntax

```
TRACe<x>:IQVTime[:MEASview<y>]:FUNCtion { NORMal | AVERage | MAXHold | MINHold }
TRACe<x>:IQVTime[:MEASview<y>]:FUNCtion?
```

Arguments

<x> = 1 to 3; Trace 4 (math trace) and Trace 5 (spectrogram) are invalid.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

NORMal selects the normal waveform display.

AVERage selects the Average display that indicates the average signal level at each time point.

MAXHold selects the Max Hold display that indicates the maximum signal level at each time point.

MINHold selects the Min Hold display that indicates the minimum signal level at each time point.

Examples

TRACE1:IQVTIME:MEASview1:FUNCTION MAXHold displays the Max Hold trace in the IQ level versus Time measurement.

TRACe<x>:IQVTime[:MEASview<y>]:SELect:I

Determines whether or not to select the I trace to obtain the maximum and minimum measurement results.

Conditions

Measurement views: RF I & Q versus Time

Group

Trace commands

Syntax

```
TRACe<x>:IQVTime[:MEASview<y>]:SELect:I { OFF | ON | 0 | 1 }
TRACe<x>:IQVTime[:MEASview<y>]:SELect:I?
```

Related Commands

TRACe<x>:IQVTime[:MEASview<y>]:SELect:Q (on page 2595)

Arguments

<x> = 1 to 3; Trace 4 (math trace) and Trace 5 (spectrogram) are invalid.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

OFF or 0 deselects the I trace.

ON or 1 selects the I trace.

Executing TRACe<x>:IQVTime[:MEASview<y>]:SELect:I ON sets TRACe<x>:IQVTime[:MEASview<y>]:SELect:Q OFF.

Examples

TRACE1:IQVTIME:MEASview1:SELECT:I ON selects the I trace in the RF I & Q versus Time measurement.

TRACe<x>:IQVTime[:MEASview<y>]:SELect:Q

Determines whether or not to select the Q trace to obtain the maximum and minimum measurement results.

Conditions

Measurement views: RF I & Q versus Time

Group

Trace commands

Syntax

```
TRACe<x>:IQVTime[:MEASview<y>]:SELect:Q { OFF | ON | 0 | 1 }
TRACe<x>:IQVTime[:MEASview<y>]:SELect:Q?
```

Related Commands

TRACe<x>:IQVTime[:MEASview<y>]:SELect:I (on page 2594)

Arguments

<x> = 1 to 3; Trace 4 (math trace) and Trace 5 (spectrogram) are invalid.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

OFF or 0 deselects the Q trace.

ON or 1 selects the Q trace.

Executing `TRACe<X>:IQVTime[:MEASview<y>]:SELect:Q ON` sets `TRACe<x>:IQVTime[:MEASview<y>]:SELect:I OFF`.

Examples

`TRACE1:IQVTIME:MEASview1:SELECT:Q ON` selects the Q trace in the RF I&Q versus Time measurement.

TRACe:MERRor[:MEASview<y>]:MODE

Sets or queries whether to display the Magnitude error trace as vectors or symbols (points).

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement view: Magnitude error

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Magnitude error versus Time).

Group

Trace commands

Syntax

```
TRACe:MERRor[:MEASview<y>]:MODE { VECTors | SYMBols }
TRACe:MERRor[:MEASview<y>]:MODE?
```

Arguments

`VECTors` displays the trace in a format that uses lines to connecting points in the trace display.

`SYMBols` displays the trace in a format that does not use lines to connect points in the trace display.

Examples

`TRACE:MERRor:MEASview2:MODE VECTORS` sets the display to connect points in the display for measview 2 of MERRor versus Time.

TRACe:OBW:MAXHold

Determines whether or not to perform a Max Hold on the spectrum data for the Occupied Bandwidth trace.

Conditions

Measurement views: Occupied Bandwidth

Group

Trace commands

Syntax

```
TRACe:OBW:MAXHold { OFF | ON | 0 | 1 }
TRACe:OBW:MAXHold?
```

Arguments

`OFF` or 0 does not perform a Max Hold on the spectrum data.

`ON` or 1 performs a Max Hold on the spectrum data.

Examples

`TRACE:OBW:MAXHOLD ON` performs a Max Hold on the spectrum data for the Occupied Bandwidth trace.

TRACe:OFDM:CONSte

Determines whether or not to show the trace in the OFDM Constellation view.

Conditions

Measurement views: OFDM

Group

Trace commands

Syntax

```
TRACe:OFDM:CONSte { OFF | ON | 0 | 1 }
TRACe:OFDM:CONSte?
```

Arguments

`OFF` or 0 hides the trace in the view.

`ON` or 1 shows the trace in the view.

Examples

`TRACE:OFDM:CONSTE ON` shows the trace in the OFDM Constellation view

TRACe:OFDM:CONSte:FREeze

Determines whether or not to freeze the display of the trace in the OFDM Constellation view.

Conditions

Measurement views: OFDM

Group

Trace commands

Syntax

```
TRACe:OFDM:CONSte:FREeze { OFF | ON | 0 | 1 }
TRACe:OFDM:CONSte:FREeze?
```

Arguments

`OFF` or 0 updates the display of the trace normally.

`ON` or 1 stops updating the display of the trace.

Examples

`TRACE:OFDM:CONSTE:FREEZE ON` freezes the trace in the OFDM Constellation view.

TRACe:OFDM:FLATness:FREeze

Sets or queries whether to freeze the average trace on the OFDM Spectral Flatness display. When the freeze state is turned on, the trace stops updating.

Conditions

Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group

Trace commands

Syntax

```
TRACe:OFDM:FLATness:FREeze {0|1|OFF|ON}
TRACe:OFDM:FLATness:FREeze?
```

Arguments

`ON` or 1 turns on the freeze state, causing the trace to stop updating.

`OFF` or 0 turns off the freeze state.

Returns

Outputs the frozen status of the selected trace as an integral value. 1 indicates a frozen trace and 0 a non-frozen trace.

Examples

`TRAC:OFDM:FLAT:FRE 1` freezes the trace, causing it to stop updating.

TRACe:OFDM:FLATness:SHOW

Displays or hides the trace on the OFDM Spectral Flatness display.

Conditions

Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group

Trace commands

Syntax

```
TRACe:OFDM:FLATness:SHOW {0|1|OFF|ON}
```

Arguments

`ON` or 1 displays the trace.

`OFF` or 0 hides the trace.

Examples

`TRACE:OFDM:FLAT:SHOW 1` turns on the display of the trace.

TRACe:PERRor[:MEASview<y>]:MODE

Sets or queries whether to display the Frequency deviation versus Time trace as vectors or symbols (points).

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement view: Phase error versus Time

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Phase error versus Time).

Group

Trace commands

Syntax

```
TRACe:PERRor[:MEASview<y>]:MODE { VECTors | SYMBols }
TRACe:PERRor[:MEASview<y>]:MODE?
```

Arguments

`VECTors` displays the trace in a format that uses lines to connecting points in the trace display.

`SYMBols` displays the trace in a format that does not use lines to connect points in the trace display.

Examples

`TRACe:PERRor:MEASview2:MODE VECTors` sets the display to connect points in the display display for measview 2 of Phase error versus Time .

TRACe<x>:PHVTime[:MEASview<y>]

Determines whether or not to show the trace in the Phase versus Time view.

Conditions

Measurement views: Phase versus Time

Group

Trace commands

Syntax

```
TRACe<x>:PHVTime[:MEASview<y>] { OFF | ON | 0 | 1 }
TRACe<x>:PHVTime[:MEASview<y>]?
```

Arguments

`OFF` or `0` hides the trace in the Phase versus Time view.

`ON` or `1` shows the trace in the Phase versus Time view.

<x> = 1 to 7; Trace 2, Trace 3 and Trace 4 (math trace) are invalid.

- Trace5: Bitmap Trace
- Trace6: Ogram
- Trace7: Ogram Line

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Examples

`TRACE1:PHVTIME:MEASview1 ON` shows the trace in the Phase versus Time view.

TRACe<x>:PHVTime[:MEASview<y>]:AVERage:COUNt

Sets or queries the number of traces to combine. This command is effective when you select `AVERage` with the `TRACe<x>:PHVTime[:MEASview<y>]:FUNCtion` command.

Conditions

Measurement views: Phase versus Time

Group

Trace commands

Syntax

```
TRACe<x>:PHVTime[:MEASview<y>]:AVERage:COUNt <number>
TRACe<x>:PHVTime[:MEASview<y>]:AVERage:COUNt?
```

Arguments

`<x>` = 1 to 3; Trace 4 (math trace) and Trace 5 (spectrogram) are invalid.

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

`<number>::<NR1>` specifies the number of traces to combine for averaging.

Range: 1 to 10000.

Examples

`TRACE1:PHVTIME:MEASview1:AVERAGE:COUNT 64` sets the average count to 64.

TRACe<x>:PHVTime[:MEASview<y>]:COLor:CURVe

Sets or queries how colors are mapped to the signal density in the Phase versus Time bitmap display. The mapping can be linear (Curve = 1), or it can be set to concentrate the resolution on the lower level of the range (Curve > 1) or the mapping can be set to show the best resolution on the upper range of density or hit count (Curve = 0.1 - 0.99). This command is only valid for the Bitmap display.

The parameter <x> = 5; only Trace 5 is valid.

Conditions

Measurement views: Phase versus Time

Group

Trace commands

Syntax

`TRACe<x>:PHVTime[:MEASview<y>]:COLor:CURVe <value>`

`TRACe<x>:PHVTime[:MEASview<y>]:COLor:CURVe?`

Arguments

`<value>` = <NRf> specifies how colors are mapped to the signal density. Range: 0.1 to 10 (the default value is 2).

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. The default value of y is 1.

Examples

TRACe5:PHVTIME:COLor:CURVe 1.5 concentrates the resolution on the lower level of the range on the Bitmap display.

TRACe<x>:PHVTime[:MEASview<y>]:COLor:SCALe:AUTO

Automatically adjusts the Max and Min color settings to display the broadest range of colors in the Phase versus Time bitmap display. This command is only valid for the Bitmap display.

The parameter <x> = 5; only Trace 5 is valid.

Conditions

Measurement views: Phase versus Time

Group

Trace commands

Syntax

TRACe<x>:PHVTime[:MEASview<y>]:COLor:SCALe:AUTO

Arguments

<x> = 5 specifies the auto-color setting for the Bitmap trace.

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. The default value of y is 1.

Examples

TRACe5:PHVTIME:COLor:SCALe:AUTO autosets the Bitmap display color scheme.

TRACe<x>:PHVTime[:MEASview<y>]:COUNt

Sets or queries how many acquisitions run in the single acquisition mode for the Max or Min Hold trace in the Phase versus Time measurement. This command is effective when TRACe<x>:PHVTime[:MEASview<y>]:FUNCtion is set to MAXHold or MINHold and INITiate:CONTinuous is set to OFF.

Conditions

Measurement views: Phase versus Time

Group

Trace commands

Syntax

```
TRACe<x>:PHVTime[:MEASview<y>]:COUNt <number>
TRACe<x>:PHVTime[:MEASview<y>]:COUNt?
```

Arguments

<x> = 1 to 3; Trace 4 (math trace) and Trace 5 (spectrogram) are invalid.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

<number>::<NR1> specifies the count for Max/Min Hold.

Range: 1 to 10000.

Examples

`TRACE1:PHVTIME:MEASview1:COUNT 32` sets the count to 32 for the Max/Min Hold trace.

TRACe<x>:PHVTime[:MEASview<y>]:COUNt:ENABle

Determines whether to enable or disable the count for the Max or Min Hold trace in the Phase versus Time view. This command is effective when `TRACe<x>:PHVTime:MEASview<y>:FUNCtion` is set to `MAXHold` or `MINHold`.

Conditions

Measurement views: Phase versus Time

Group

Trace commands

Syntax

```
TRACe<x>:PHVTime[:MEASview<y>]:COUNt:ENABle { OFF | ON | 0 | 1 }
TRACe<x>:PHVTime[:MEASview<y>]:COUNt:ENABle?
```

Related Commands

Arguments

`<x>` = 1 to 3; Trace 4 (math trace) and Trace 5 (spectrogram) are invalid.

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

`OFF` or 0 disables the count for the Max/Min Hold trace.

`ON` or 1 enables the count for the Max/Min Hold trace.

Examples

`TRACE1:PHVTIME:MEASview1:COUNT:ENABLE ON` enables the Max/Min Hold count.

TRACe<x>:PHVTime[:MEASview<y>]:COUNt:RESet (No Query Form)

Clears the Max or Min Hold data and counter, and restarts the process in the Phase versus Time view. This command is effective when `TRACe<x>:PHVTime:MEASview<y>:FUNCtion` is set to `MAXHold` or `MINHold`.

Conditions

Measurement views: Phase versus Time

Group

Trace commands

Syntax

```
TRACe<x>:PHVTime[:MEASview<y>]:COUNt:RESet
```

Arguments

`<x>` = 1 to 3; Trace 4 (math trace) and Trace 5 (spectrogram) are invalid.

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

Examples

`TRACE1:PHVTIME:MEASview1:COUNT:RESET` clears the Max/Min Hold data and counter, and restarts the process.

TRACe<x>:PHVTime[:MEASview<y>]:DETection

Sets or queries the display detector (method to be used for decimating traces to fit the available horizontal space on screen). The number of horizontal pixels on screen is generally smaller than that of waveform data points. When actually displayed, the waveform data is therefore thinned out, according to the number of pixels, for being compressed.

Conditions

Measurement views: Phase versus Time

Group

Trace commands

Syntax

```
TRACe<x>:PHVTime[:MEASview<y>]:DETection { AVERage | POSitive | NEGative | POSNegative | SAMPle }
TRACe<x>:PHVTime[:MEASview<y>]:DETection?
```

Arguments

`<x>` = 1 to 6.

- Trace 4: Math trace (not applicable)
- Trace 5: Bitmap trace (not applicable)
- Trace 6: Ogram

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

`AVERage` displays the average data value for each pixel.

`POSitive` displays the maximum data value for each pixel.

`NEGative` displays the minimum data value for each pixel.

`POSNegative` displays the maximum and minimum data values for each pixel.

`SAMPle` displays the first sample value received for each pixel.

Examples

`TRACe1:PHVTime[:MEASview<y>]:DETection POSitive` displays the maximum data value for each pixel.

TRACe<x>:PHVTime[:MEASview<y>]:DOT:PERSistent

Determines whether to enable or disable the dot persistence for the bitmap trace (Trace 5) in the Phase versus Time view.

The parameter <x> = 5; Only Trace 5 (bitmap trace) is valid.

Conditions

Measurement views: Phase versus Time

Group

Trace commands

Syntax

`TRACe<x>:PHVTime[:MEASview<y>]:DOT:PERSistent { OFF | ON | 0 | 1 }`

`TRACe<x>:PHVTime[:MEASview<y>]:DOT:PERSistent?`

Arguments

`OFF` or `0` disables the dot persistence.

`ON` or `1` enables the dot persistence.

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`TRACE5:PHVTIME:DOT:PERSISTENT ON` enables the dot persistence in the Phase versus Time view.

TRACe<x>:PHVTime[:MEASview<y>]:DOT:PERSistent:TYPE

Sets or queries the persistence type for the bitmap trace (Trace 5) in the Phase versus Time view.

The parameter <x> = 5; Only Trace 5 (bitmap trace) is valid.

Conditions

Measurement views: Phase versus Time

Group

Trace commands

Syntax

`TRACe<x>:PHVTime[:MEASview<y>]:DOT:PERSistent:TYPE { VARiable | INFinite }`

`TRACe<x>:PHVTime[:MEASview<y>]:DOT:PERSistent:TYPE?`

Arguments

`VARiable`: Selects the variable persistence display which leaves acquired data points on the display for a period of time specified by the `TRACe<x>:PHVTime[:MEASview<y>]:DOT:PERSistent:VARiable` command.

`INFinite`: Selects the infinite persistence display which accumulates data points on the display indefinitely.

The parameter <y>= 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`TRACE5:PHVTIME:DOT:PERSISTENT:TYPE VARiable` selects the variable persistence display.

TRACe<x>:PHVTime[:MEASview<y>]:DOT:PERSistent:VARiable

Sets or queries how long data points are displayed. This command is effective when `TRACe<x>:PHVTime[:MEASview<y>]:DOT:PERSistent:TYPE` is set to `VARiable`. This affects the display only.

The parameter <x> = 5; Only Trace 5 (bitmap trace) is valid.

Conditions

Measurement views: Phase versus Time

Group

Trace commands

Syntax

`TRACe<x>:PHVTime[:MEASview<y>]:DOT:PERSistent:VARiable <number>`

`TRACe<x>:PHVTime[:MEASview<y>]:DOT:PERSistent:VARiable?`

Arguments

`<number>` = <NRf> specifies the how long waveform points are displayed on the screen. Range: 0 s to 1000 s (the default value is 10 ms).

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`TRACE5:PHVTIME:DOT:PERSISTENT:VARIABLE 20` specifies that the waveform points are displayed on the screen for a period of 20 before they disappear.

TRACe<x>:PHVTime[:MEASview<y>]:FREeze

Determines whether to freeze the trace display in the Phase versus Time measurement.

Conditions

Measurement views: Phase versus Time

Group

Trace commands

Syntax

```
TRACe<x>:PHVTime[:MEASview<y>]:FREeze { OFF | ON | 0 | 1 }
TRACe<x>:PHVTime[:MEASview<y>]:FREeze?
```

Arguments

`<x>` = 1 to 7.

- Trace 4: Math trace
- Trace 5: Bitmap trace
- Trace 6: Ogram
- Trace 7: Ogram Line

`<y>` = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

`OFF` or 0 updates the trace display normally.

`ON` or 1 stops updating trace display.

Examples

`TRACE1:PHVTIME:MEASview1:FREEZE ON` stops updating trace display.

TRACe<x>:PHVTime[:MEASview<y>]:FUNCtion

Sets or queries the trace function in the Phase versus Time measurement.

Conditions

Measurement views: Phase versus Time

Group

Trace commands

Syntax

```
TRACe<x>:PHVTime[:MEASview<y>]:FUNCtion { NORMal | AVERage | MAXHold | MINHold }
TRACe<x>:PHVTime[:MEASview<y>]:FUNCtion?
```

Arguments

<x> = 1 to 3; Trace 4 (math trace) and Trace 5 (spectrogram) are invalid.

<y> = 1 to 8; represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value 1.

NORMal selects the normal waveform display.

AVERage selects the Average display that indicates the average phase drift at each time point.

MAXHold selects the Max Hold display that indicates the maximum phase drift at each time point.

MINHold selects the Min Hold display that indicates the minimum phase drift at each time point.

POSNegative displays the maximum and minimum data values for each pixel.

SAMPle displays the first sample value received for each pixel.

Examples

TRACE1:PHVTIME:MEASview1:FUNCTION MAXHold displays the Max Hold trace in the Phase versus Time measurement.

TRACe<x>:PHVTime[:MEASview<y>]:SELect

Sets or queries the selected trace.

Conditions

Applicable measurement view must be active.

Group

Trace commands

Syntax

```
TRACe<x>:PHVTime[:MEASview<y>]:SELect {ON | OFF | 0 | 1}
TRACe<x>:PHVTime[:MEASview<y>]:SELect?
```

Arguments

ON or 1 selects the specified trace.

The parameter <x> = 1 to 7.

- Trace5: Bitmap Trace
- Trace6: Ogram
- Trace7: Ogram Line

The parameter <y>=1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional by default y is 1

Returns

`0` indicates the specified trace is not the selected trace.

`1` indicates the specified trace is the selected trace.

Examples

`TRACe6:PHVTime:MEASview1:SELect 1` selects Trace6 as the selected trace for measview 1 of Phase vs time.

TRACe<x>:PHVTime[:MEASview<y>]:SELect:LINe

Sets or queries the line number to send to the Phase vs Time ogram display.

Conditions

Applicable measurement view must be active.

Group

Trace commands

Syntax

`TRACe<x>:PHVTime[:MEASview<y>]:SELect:LINe <number>`

`TRACe<x>:PHVTime[:MEASview<y>]:SELect:LINe?`

Related Commands

TRACe<x>:PHVTime[:MEASview<y>] (on page 2599)

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional. By default y is 1.

<x> is 6 for ogram trace

`<number>::=<NR1>` specifies the line number to send to the display.

Range: 0 to the maximum line number of the ogram displayed on screen

Returns

`<NRf>` specifies the line number to send to the spectrum display.

Examples

`TRACe6:PHVTime:MEASview1:SELect:LINe 75` selects Line #75 in the ogram to send to the time plot display.

TRACe:SEM:COUNt

Sets or queries how many acquisitions run in the single acquisition mode for multi-trace functions (Max Hold and Average) in the Spectral Emissions Mask measurement. This command is effective when `TRACe:SEM:FUNCtion` is set to `MAXHold` or `AVERage` and `INITiate:CONTinuous` is set to OFF.

Conditions

Measurement views: Spectral Emissions Mask

Group

Trace commands

Syntax

```
TRACe:SEM:COUNt <number>
TRACe:SEM:COUNt?
```

Arguments

`<number>::<NR1>` specifies the count for multi-trace functions.

Range: 1 to 10000.

Examples

`TRACE:SEM:COUNT 32` sets the count to 32 for multi-trace functions.

TRACe:SEM:COUNt:RESet (No Query Form)

Clears the multi-function (Max Hold or Average) data and counter and restarts the process in the Spectral Emissions Mask view. This command is effective when `TRACe:SEM:FUNCtion` is set to `MAXHold` or `AVERage`.

Conditions

Measurement views: Spectral Emissions Mask

Group

Trace commands

Syntax

```
TRACe:SEM:COUNt:RESet
```

Arguments

None

Examples

`TRACE:SEM:COUNT:RESET` clears the multi-function data and counter, and restarts the process.

TRACe:SEM:FUNCtion

Sets or queries the trace function in the Spectral Emissions Mask view.

Conditions

Measurement views: Spectral Emissions Mask

Group

Trace commands

Syntax

```
TRACe:SEM:FUNCtion { NONE | MAXHold | AVERage }
TRACe:SEM:FUNCtion?
```

Arguments

`NONE` selects normal display.

MAXHold selects the Max Hold display that indicates the maximum amplitude drift at each frequency point.

AVERage selects the Average display that indicates the average amplitude drift at each frequency point.

Examples

TRACE:SEM:FUNCTION MAXHold displays the Max Hold trace in the Spectral Emissions Mask measurement.

TRACe:SGRam[:MEASview<y>]:DETection

Sets or queries the display detector (method to be used for decimating traces to fit the available horizontal space on screen). The number of horizontal pixels on screen is generally smaller than that of waveform data points. When actually displayed, the waveform data is therefore thinned out, according to the number of pixels, for being compressed.

Conditions

Measurement views: Spectrogram

Group

Trace commands

Syntax

```
TRACe:SGRam[:MEASview<y>]:DETection { AVERage | POSitive | NEGative | CAVerage | CPEak | QUASipeak
    | AVGLog }
TRACe:SGRam[:MEASview<y>]:DETection?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional The default value of y is 1.

AVERage displays the average data value for each pixel.

POSitive displays the maximum data value for each pixel.

NEGative displays the minimum data value for each pixel.

CAVerage displays the CISPR average value for each pixel.

CPEak displays the CISPR peak value for each pixel.

QUASipeak displays the quasi-peak value for each pixel.

AVGLog displays the average data value of logs for each pixel.

Returns

{AVERage | POSitive | NEGative | CAVerage | CPEak | QUASipeak | AVGLog}

Examples

TRACe:SGRam:MEASview1:DETection POSitive displays the maximum data value for each pixel.

TRACe:SGRam[:MEASview<y>]:FREeze

Determines whether or not to freeze the spectrogram display.

Conditions

Measurement views: Spectrogram

Group

Trace commands

Syntax

```
TRACe:SGRam[:MEASview<y>]:FREeze { OFF | ON | 0 | 1 }
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional. The default value of y is 1.

OFF or 0 updates the display of the spectrogram normally.

ON or 1 stops updating the display of the spectrogram.

Examples

TRACe:SGRam:MEASview1:FREeze ON freezes the spectrogram display.

TRACe:SGRam[:MEASview<y>]:FUNCtion

NOTE: This function is no longer available in software versions 2.3.0159 and later. Sending this command selects the correct. SENSe:SGRam[:MEASview<y>]:TIME[:SCALe]:MODe and TRACe:SGRam[:MEASview<y>]:DETection to achieve the same result as with previous software versions. The query will return the equivalent trace function based on the MODe and DETection settings.

Sets or queries the trace function for the specified trace in the spectrogram.

Conditions

Measurement views: Spectrogram

Group

Trace commands

Syntax

```
TRACe:SGRam[:MEASview<y>]:FUNCtion { NONE | AVERage | MAXHold | MINHold| AVGLog }
TRACe:SGRam[:MEASview<y>]:FUNCtion?
```

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional. The default value of y is 1.

NONE selects the normal spectrogram display.

AVERage selects the Average display that indicates the average signal level at each frequency point.

MAXHold selects the Max Hold display that indicates the maximum signal level at each frequency point.

MINHold selects the Min Hold display that indicates the minimum signal level at each frequency point.

AVGLog selects the Average of logs display that indicates the average signal level of logs at each frequency point.

Returns

{NONE | AVERage | MAXHold | MINHold| AVGLog}

Examples

TRACe:SGRam:MEASview1:FUNCtion MAXHold selects the Max Hold display for the spectrogram.

TRACe:SGRam[:MEASview<y>]:FUNCtion:TIME

NOTE: This command has a new function in software versions 2.3.159 and above. Sets the Spectrum Monitor mode values for Time per update. Sending this command sets `SENSe:SGRam[:MEASview<y>]:TIME:SCALe:PER:UPDate:MINutes` and `SENSe:SGRam[:MEASview<y>]:WATerfall:DIRection` based on the value sent in. The query form returns the number of minutes based on the new settings values.

Conditions

Measurement views: Spectrogram

Group

Trace commands

Syntax

```
TRACe:SGRam[:MEASview<y>]:FUNCtion:TIME <value>
TRACe:SGRam[:MEASview<y>]:FUNCtion:TIME?
```

Arguments

`<y>` = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional. By default y is 1.

`<value>::<NR1>` specifies the time length in Spectrum Monitor mode.

Range: 1 to 600 minutes and 60 seconds.

Returns

`<NRf>` specifies the time length in Spectrum Monitor mode.

Examples

`TRACe:SGRam:MEASview1:FUNCtion:TIME 6E2` sets the Time/update value to 600 minutes.

TRACe:SGRam[:MEASview<y>]:RESet

Determines whether or not to reset the spectrogram display.

Conditions

Measurement view: Spectrogram

Group

Trace commands

Syntax

```
TRACe:SGRam[:MEASview<y>]:RESet {OFF | ON | 0 | 1}
```

Arguments

`<y>` = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

`OFF` or 0 does not reset the display of the spectrogram.

`ON` or 1 resets the display of the spectrogram.

Examples

TRACe:SGRam:MEASview1:RESet ON resets the spectrogram measurement view 1 display.

TRACe:SGRam[:MEASview<y>]:SELect:LINE

Sets or queries the line number to send to the spectrum display.

Conditions

Measurement views: Spectrogram

Group

Trace commands

Syntax

```
TRACe:SGRam[:MEASview<y>]:SELect:LINE <number>
TRACe:SGRam[:MEASview<y>]:SELect:LINE?
```

Related Commands

TRACe<x>:SPECtrum (on page 2684)

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional by default y is 1.

<number> ::= <NR1> specifies the line number to send to the spectrum display.

Range: 0 to the maximum line number of the spectrogram displayed on screen.

Returns

<NRf> specifies the line number to send to the spectrum display.

Examples

TRACe:SGRam:MEASview1:SELect:LINE 75 selects Line #75 in the spectrogram to send to the spectrum display.

TRACe:SPURious:COUNt

Sets or queries how many acquisitions run in the single acquisition mode for multi-trace functions (Max Hold and Average) in the Spurious measurement. This command is effective when TRACe:SPURious:FUNCtion is set to MAXHold or AVERage and INITiate:CONTinuous is set to OFF.

Conditions

Measurement views: Spurious

Group

Trace commands

Syntax

```
TRACe:SPURious:COUNt <number>
TRACe:SPURious:COUNt?
```

Arguments

<number>::<NR1> specifies the count for multi-trace functions.

Range: 1 to 10000.

Examples

`TRACE:SPURIOUS:COUNT 32` sets the count to 32 for multi-trace functions.

TRACe:SPURious:COUNt:ENABle

Determines whether to enable or disable the count for multi-trace functions (Max Hold and Average) in the Spurious view. This command is effective when `TRACe:SPURious:FUNCtion` is set to `MAXHold` or `AVERage`.

Conditions

Measurement views: Spurious

Group

Trace commands

Syntax

```
TRACe:SPURious:COUNt:ENABle { OFF | ON | 0 | 1 }
TRACe:SPURious:COUNt:ENABle?
```

Arguments

`OFF` or 0 disables the count for multi-trace functions.

`ON` or 1 enables the count for multi-trace functions.

Examples

`TRACE:SPURIOUS:COUNT:ENABLE ON` enables the count for multi-trace functions.

TRACe:SPURious:COUNt:RESet (No Query Form)

Clears the multi-function (Max Hold or Average) data and counter, and restarts the process in the Spurious view. This command is effective when `TRACe:SPURious:FUNCtion` is set to `MAXHold` or `AVERage`.

Conditions

Measurement views: Spurious

Group

Trace commands

Syntax

```
TRACe:SPURious:COUNt:RESet
```

Arguments

None

Examples

`TRACE:SPURIOUS:COUNT:RESET` clears the multi-function data and counter, and restarts the process.

TRACe:SPURious:FREeze

Determines whether or not to freeze the display of the trace in the Spurious view.

Conditions

Measurement views: Spurious

Group

Trace commands

Syntax

```
TRACe:SPURious:FREeze { OFF | ON | 0 | 1 }
TRACe:SPURious:FREeze?
```

Arguments

`OFF` or 0 updates the display of the trace normally.

`ON` or 1 stops updating the display of the trace.

Examples

`TRACE:SPURIOUS:FREEZE ON` stops updating the display of the trace.

TRACe:SPURious:FUNCtion

Sets or queries the trace function in the Spurious view.

Conditions

Measurement views: Spurious

Group

Trace commands

Syntax

```
TRACe:SPURious:FUNCtion { NONE | MAXHold | AVERage | AVGLog}
TRACe:SPURious:FUNCtion?
```

Arguments

`NONE` selects normal display.

`MAXHold` selects the Max Hold display that indicates the maximum amplitude drift at each frequency point.

`AVERage` selects the Average display that indicates the average amplitude drift at each frequency point.

`AVGLog` selects the Average of logs display that indicates the average amplitude drift of logs at each frequency point.

Examples

`TRACE:SPURIOUS:FUNCTIONMAXHold` displays the Max Hold trace in the Spurious measurement.

TRACe:WLAN:AD:CONSte:TYPE

Sets or queries the trace type in the 802.11ad Constellation display.

Conditions

Measurement view: 802.11ad Constellation

Group

Trace commands

Syntax

```
TRACe:WLAN:AD:CONSte:TYPE { ROTA | REGU }
TRACe:WLAN:AD:CONSte:TYPE?
```

Arguments

ROTA is pi/2 Rotated trace type

REGU is the Regular trace type

Returns

ROTA means the trace type is set to pi/2 rotated

REGU means the trace type is set to regular

Examples

TRACE:WLAN:AD:CONSTE:TYPE ROTA sets the trace type to PI/2 Rotated.

TRACe:WLAN:AY:CONSte:TYPE

Set or query the Trace Type (Regular or PI/2 Rotated).

Conditions

Measurement view: 802.11ay Analysis Constellation

Group

Trace commands

Syntax

```
TRACe:WLAN:AY:CONSte:TYPE { REGU | ROTA }
TRACe:WLAN:AY:CONSte:TYPE?
```

Arguments

REGU sets the Trace Type as Regular.

ROTA sets the Trace Type as PI/2 Rotated.

Returns

REGU means that the Trace Type is Regular.

ROTA means that the Trace Type is PI/2 Rotated

Examples

TRAC:WLAN:AY:CONS:TYPE REGU sets the Trace Type to Regular.

TRAC:WLAN:AY:CONS:TYPE? may return ROTA, which means the Trace Type is PI/2 Rotated.

TRACe:WLAN:FLATness:FREeze

Sets or queries whether to freeze the average trace on the WLAN Spectral Flatness display. When the freeze state is turned on, the selected trace stops updating.

NOTE: WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Trace commands

Syntax

```
TRACe:WLAN:FLATness:FREeze {0|1|OFF|ON}
TRACe:WLAN:FLATness:FREeze?
```

Arguments

`ON` or 1 turns on the freeze state, causing the selected trace to stop updating.

`OFF` or 0 turns off the freeze state.

Returns

Outputs the frozen status of the selected trace as an integral value.

1 indicates a frozen trace and 0 a non-frozen trace.

Examples

`TRAC:WLAN:FLAT:FRE 1` freezes the trace, causing it to stop updating.

TRACe:WLAN:FLATness:SHOW

Displays or hides the trace on the WLAN Spectral Flatness display.

NOTE: WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Trace commands

Syntax

```
TRACe:WLAN:FLATness:SHOW {0|1|OFF|ON}
TRACe:WLAN:FLATness:SHOW?
```

Arguments

`ON` or 1 displays the trace.

`OFF` or 0 hides the trace.

Examples

`TRACE:WLAN:FLAT:SHOW 1` turns on the display of the trace.

TRACe1:TOVerview[:MEASview<y>]

Enables display of or queries the display status of the specified trace. Only Trace1 is valid.

Conditions

Measurement views: General Waveform display

Group

Trace commands

Syntax

```
TRACe1:TOVerview[:MEASview<y>] { OFF | ON | 0 | 1 }
TRACe1:TOVerview[:MEASview<y>]?
```

Arguments

`OFF` or 0 disables the trace overview.

`ON` or 1 enables the trace overview.

Examples

`TRACe1:TOVerview:MEASVIEW1 ON` shows the trace1 in the Time overview Measurement View 1.

TRACe1:TOVerview[:MEASview<y>]:AVERage:COUNt

Sets or queries the number of traces averaged to generate the specified trace. Only Trace1 is valid.

Conditions

Measurement views: Trace Function set to Average

Group

Trace commands

Syntax

```
TRACe1:TOVerview[:MEASview<y>]:AVERage:COUNt <value>
TRACe1:TOVerview[:MEASview<y>]:AVERage:COUNt?
```

Arguments

`<value>::<NR1>` an integer number of traces to average to create the waveform display.

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<NRf>`

Examples

`TRACe1:TOVerview:MEASview1:AVERage:COUNt 200` sets the Average count for Trace 1 to 200.

TRACe1:TOVerview[:MEASview<y>]:COUNt

Enables or queries the count set for the specified trace. Only Trace1 is valid.

Conditions

Measurement views: Trace Function set to Average

Group

Trace commands

Syntax

```
TRACe1:TOVerview[:MEASview<y>]:COUNt <value>
TRACe1:TOVerview[:MEASview<y>]:COUNt?
```

Arguments

`<value>::<NRf>`

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Returns

`<NRf>`

Examples

`TRACe1:TOVerview:MEASview1:COUNt 200` sets the Trace1 Average count to 200.

TRACe1:TOVerview[:MEASview<y>]:COUNt:ENABle

Enables or queries the Average count for the specified trace. Only Trace1 is valid.

Conditions

Measurement views: Trace Function set to Average

Group

Trace commands

Syntax

```
TRACe1:TOVerview[:MEASview<y>]:COUNt:ENABle { OFF | ON | 0 | 1 }
TRACe1:TOVerview[:MEASview<y>]:COUNt:ENABle?
```

Arguments

`OFF` or 0 disables the trace count.

`ON` or 1 enables the trace count.

Examples

`TRACe1:TOVerview:MEASview1:COUNt:ENABle ON` enables the Trace1 Average count.

TRACe1:TOVerview[:MEASview<y>]:COUNt:RESet (No Query Form)

Resets the waveform count for the specified trace. Only Trace1 is valid.

Conditions

Measurement views: Trace Function set to Average

Group

Trace commands

Syntax

```
TRACe1:TOVerview[:MEASview<y>]:COUNt:RESet
```

Arguments

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. Default value is 1.

Examples

`TRACe1:TOVerview:MEASview1:COUNt:RESet` sets the trace count to 1.

TRACe1:TOVerview[:MEASview<y>]:DETection

Enables or queries the type of detection for the specified trace. Only Trace1 is valid.

Conditions

Measurement views: all

Group

Trace commands

Syntax

```
TRACe1:TOVerview[:MEASview<y>]:DETection { AVERage | POSitive | NEGative | POSNegative | SAMPle }
TRACe1:TOVerview[:MEASview<y>]:DETection?
```

Arguments

`AVERage` displays the average data value for each pixel.

`POSitive` displays the maximum data value for each pixel.

`NEGative` displays the minimum data value for each pixel.

`POSNegative` displays the maximum and minimum data values for each pixel.

`SAMPle` displays the first sample value received for each pixel.

Examples

`TRACe1:TOVerview:MEASview1:DETection POSitive` enables positive detection on Trace1.

TRACe1:TOVerview[:MEASview<y>]:FREeze

Enables or queries a halt to acquisition updates for the specified trace. Only Trace1 is valid.

Conditions

Measurement views: all

Group

Trace commands

Syntax

```
TRACe1:TOVerview[:MEASview<y>]:FREeze { OFF | ON | 0 | 1 }
TRACe1:TOVerview[:MEASview<y>]:FREeze?
```

Arguments

`OFF` or 0 disables the trace freeze function.

`ON` or 1 enables the trace freeze function.

Examples

`TRACe1:TOVerview:MEASview1:FREeze ON` Halts acquisition updates to Trace1.

TRACe1:TOVerview[:MEASview<y>]:FUNCTion

Enables or queries the selected Function for the specified trace. Only Trace1 is valid.

Conditions

Measurement views: all

Group

Trace commands

Syntax

```
TRACe1:TOVerview[:MEASview<y>]:FUNCTion { NORMal | AVERage | MAXHold | MINHold }
TRACe1:TOVerview[:MEASview<y>]:FUNCTion?
```

Arguments

`NORMal` Each new trace is displayed and then replaced by the next trace.

`AVERage` Multiple traces are averaged together to generate the displayed trace.

`MAXHold` Displays the maximum value in the trace record for each display point.

`MINHold` Displays the minimum value in the trace record for each display point.

Examples

`TRACe1:TOVerview:MEASview1:FUNCTion AVERage` enables the averaging of multiple traces on Trace1.

TRACe<x>:{AM|FM|PM}

Determines whether or not to show the trace in the AM/FM/PM view.

Conditions

Measurement views: AM, FM, PM

Group

Trace commands

Syntax

```
TRACe<x>:{AM|FM|PM} { OFF | ON | 0 | 1 }
TRACe<x>:{AM|FM|PM}?
```

Arguments

`OFF` or 0 hides the trace in the AM, FM, or PM view.

`ON` or 1 shows the trace in the AM, FM, or PM view.

Examples

`TRACE:AM ON` shows the trace in the AM view.

TRACe<x>:AVTime[:MEASview<y>]

Determines whether or not to show the specified trace in the Amplitude versus Time view.

The parameter <x> = 1 to 7; All traces are valid.

Conditions

Measurement views: Amplitude versus Time

Group

Trace commands

Syntax

```
TRACe<x>:AVTime[:MEASview<y>] { OFF | ON | 0 | 1 }
TRACe<x>:AVTime[:MEASview<y>]?
```

Arguments

`OFF` or `0` hides the trace in the Amplitude versus Time view.

`ON` or `1` shows the trace in the Amplitude versus Time view.

<x> = 1 to 7.

- Trace4: Math Trace
- Trace5: Bitmap Trace
- Trace6: Ogram
- Trace7: Ogram Line

The parameter <y>= 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`TRACE1:AVTIME:MEASVIEW1 ON` shows Trace 1 in the Amplitude versus Time Measurement View 1.

TRACe<x>:AVTime[:MEASview<y>]:AVERage:COUNt

Sets or queries the number of traces to combine. This command is effective when you select AVERage with the `TRACe<x>:AVTime[:MEASview<y>]:FUNCtion` command.

The parameter <x> = 1 to 3; Trace 4 (math trace) is invalid.

Conditions

Measurement views: Amplitude versus Time

Group

Trace commands

Syntax

```
TRACe<x>:AVTime[:MEASview<y>]:AVERage:COUNt <number>
TRACe<x>:AVTime[:MEASview<y>]:AVERage:COUNt?
```

Arguments

`<number>::<NR1>` specifies the number of traces to combine for averaging.

Range: 1 to 10000.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter <y>= 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

TRACE1:AVTIME:MEASVIEW1:AVERAGE:COUNT 64 sets the average count to 64 for Trace 1 for Amplitude versus Time Measurement View 1.

TRACe<x>:AVTime[:MEASview<y>]:AVERage:RESet (No Query Form)

Restarts acquisition and display of waveforms for the specified trace. For an Average, Max Hold, or Min Hold trace, it restarts the sequence, discarding accumulated data and resetting the counter.

The parameter <x> = 1 to 3; Trace 4 (math trace) is invalid.

Conditions

Measurement views: Amplitude versus Time

Group

Trace commands

Syntax

TRACe<x>:AVTime[:MEASview<y>]:AVERage:RESet

Related Commands

TRACe<x>:AVTime[:MEASview<y>]:FUNCtion (on page 2628)

Arguments

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter <y>= 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

TRACE1:AVTIME:MEASVIEW1:AVERAGE:RESET restarts acquisition and display of waveforms for Trace 1 on Amplitude versus Time Measurement View 1.

TRACe<x>:AVTime[:MEASview<y>]:COLor:CURVe

Sets or queries how colors are mapped to the signal density in the Amplitude versus Time bitmap display. The mapping can be linear (Curve = 1), or it can be set to concentrate the resolution on the lower level of the range (Curve > 1) or the mapping can be set to show the best resolution on the upper range of density or hit count (Curve = 0.1 - 0.99). This command is only valid for the Bitmap display.

The parameter <x> = 5; only Trace 5 is valid.

Conditions

Measurement views: Amplitude versus Time

Group

Trace commands

Syntax

`TRACe<x>:AVTime[:MEASview<y>]:COLor:CURVe <value>`

`TRACe<x>:AVTime[:MEASview<y>]:COLor:CURVe?`

Arguments

`<value>` = <NRf> specifies how colors are mapped to the signal density. Range: 0.1 to 10 (the default value is 2).

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`TRACe5:AVTIME:COLor:CURVe 1.5` concentrates the resolution on the lower level of the range on the Bitmap display.

TRACe<x>:AVTime[:MEASview<y>]:COLor:SCALe:AUTO (No Query Form)

Automatically adjusts the Max and Min color settings to display the broadest range of colors in the Amplitude versus Time bitmap display. This command is only valid for the Bitmap display.

The parameter <x> = 5; only Trace 5 is valid.

Conditions

Measurement views: Amplitude versus Time

Group

Trace commands

Syntax

`TRACe<x>:AVTime[:MEASview<y>]:COLor:SCALe:AUTO`

Arguments

`<x>` = 5 specifies the auto-color setting for the Bitmap trace.

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`TRACe5:AVTIME:COLor:SCALe:AUTO` autosets the Bitmap display color scheme.

TRACe<x>:AVTime[:MEASview<y>]:COUNt

Sets or queries how many acquisitions run in the single acquisition mode for the Max or Min Hold trace in the Amplitude versus Time measurement. This command is effective when `TRACe<x>:AVTime[:MEASview<y>]:FUNCtion` is set to `MAXHold` or `MINHold` and `INITiate:CONTinuous` is set to OFF.

The parameter <x> = 1 to 3; Trace 4 (math trace) is invalid.

Conditions

Measurement views: Amplitude versus Time

Group

Trace commands

Syntax

```
TRACe<x>:AVTime[:MEASview<y>]:COUNt <number>
TRACe<x>:AVTime[:MEASview<y>]:COUNt?
```

Arguments

`<number>::<NR1>` specifies the count for Max/Min Hold.

Range: 1 to 10000.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

The parameter <y>= 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`TRACE1:AVTIME:MEASVIEW1:COUNT 32` sets the count to `32` for Trace 1 on Amplitude versus Time Measurement View 1.

TRACe<x>:AVTime[:MEASview<y>]:COUNt:ENABle

Determines whether to enable or disable the count for the Max or Min Hold trace in the Amplitude versus Time view. This command is effective when `TRACe<x>:AVTime[:MEASview<y>]:FUNCtion` is set to `MAXHold` or `MINHold`.

The parameter <x> = 1 to 3; Trace 4 (math trace) is invalid.

Conditions

Measurement views: Amplitude versus Time

Group

Trace commands

Syntax

```
TRACe<x>:AVTime[:MEASview<y>]:COUNt:ENABle { OFF | ON | 0 | 1 }
TRACe<x>:AVTime[:MEASview<y>]:COUNt:ENABle?
```

Arguments

`OFF` or 0 disables the count for the Max/Min Hold trace.

`ON` or 1 enables the count for the Max/Min Hold trace.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

The parameter <y>= 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`TRACE1:AVTIME:MEASVIEW1:COUNT:ENABLE ON` enables the Max/Min Hold count for Trace 1 on Amplitude versus Time Measurement View 1.

TRACe<x>:AVTime[:MEASview<y>]:COUNt:RESet (No Query Form)

Clears the Max or Min Hold data and counter, and restarts the process for the specified trace in the Amplitude versus Time view. This command is effective when `TRACe<x>:AVTime[:MEASview<y>]:FUNCtion` is set to `MAXHold` or `MINHold`.

The parameter <x> = 1 to 3; Trace 4 (math trace) is invalid.

Conditions

Measurement views: Amplitude versus Time

Group

Trace commands

Syntax

```
TRACe<x>:AVTime[:MEASview<y>]:COUNt:RESet
```

Arguments

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

The parameter <y>= 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`TRACE1:AVTIME:MEASVIEW1:COUNT:RESET` clears the Max/Min Hold data and counter, and restarts the process for Trace 1 on Amplitude versus Time Measurement View 1.

TRACe<x>:AVTime[:MEASview<y>]:DETection

Sets or queries the display detector (method to be used for decimating traces to fit the available horizontal space on screen). The number of horizontal pixels on screen is generally smaller than that of waveform data points. When actually displayed, the waveform data is therefore thinned out, according to the number of pixels, for being compressed.

Conditions

Measurement views: Amplitude versus Time

Group

Trace commands

Syntax

```
TRACe<x>:AVTime[:MEASview<y>]:DETection { AVERage | POSitive | NEGative | POSNegative | SAMPle }
TRACe<x>:AVTime[:MEASview<y>]:DETection?
```

Arguments

`AVERage` displays the average data value for each pixel.

`POSitive` displays the maximum data value for each pixel.

`NEGative` displays the minimum data value for each pixel.

`POSNegative` displays the maximum and minimum data values for each pixel.

`SAMPle` displays the first sample value received for each pixel.

The parameter <x> = 1 to 6.

- Trace 4: Math trace (not applicable)
- Trace 5: Bitmap Trace (not applicable)
- Trace 6: Ogram

The parameter <y>= 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`TRACE:AVTIME:MEASVIEW1:DETECTION AVERage` specifies that the trace displays the average data value for each pixel on Amplitude versus Time Measurement View 1.

TRACe<x>:AVTime[:MEASview<y>]:DOT:PERSistent

Determines whether to enable or disable the dot persistence for the bitmap trace (Trace 5) in the Amplitude versus Time view.

The parameter <x> = 5; Only Trace 5 (bitmap trace) is valid.

Conditions

Measurement views: Amplitude versus Time

Group

Trace commands

Syntax

`TRACe<x>:AVTime[:MEASview<y>]:DOT:PERSistent { OFF | ON | 0 | 1 }`

`TRACe<x>:AVTime[:MEASview<y>]:DOT:PERSistent?`

Arguments

`OFF` or `0` disables the dot persistence.

`ON` or `1` enables the dot persistence.

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`TRACE5:AVTIME:DOT:PERSISTENT ON` enables the dot persistence in the Amplitude versus Time view.

TRACe<x>:AVTime[:MEASview<y>]:DOT:PERSistent:TYPE

Sets or queries the persistence type for the bitmap trace (Trace 5) in the Amplitude versus Time view.

The parameter <x> = 5; Only Trace 5 (bitmap trace) is valid.

Conditions

Measurement views: Amplitude versus Time

Group

Trace commands

Syntax

`TRACe<x>:AVTime[:MEASview<y>]:DOT:PERSistent:TYPE { VARiable | INFinite }`

`TRACe<x>:AVTime[:MEASview<y>]:DOT:PERSistent:TYPE?`

Arguments

`VARiable`: Selects the variable persistence display which leaves acquired data points on the display for a period of time specified by the `TRACe<x>:AVTime[:MEASview<y>]:DOT:PERSistent:VARiable` command.

INFinite: Selects the infinite persistence display which accumulates data points on the display indefinitely.

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

TRACE5:AVTIME:DOT:PERSISTENT:TYPE VARiable selects the variable persistence display.

TRACe<x>:AVTime[:MEASview<y>]:DOT:PERSistent:VARiable

Sets or queries how long data points are displayed. This command is effective when TRACe<x>:AVTime[:MEASview<y>]:DOT:PERSistent:TYPE is set to VARiable. This affects the display only.

The parameter <x> = 5; Only Trace 5 (bitmap trace) is valid.

Conditions

Measurement views: Amplitude versus Time

Group

Trace commands

Syntax

TRACe<x>:AVTime[:MEASview<y>]:DOT:PERSistent:VARiable <number>

TRACe<x>:AVTime[:MEASview<y>]:DOT:PERSistent:VARiable?

Arguments

<number> = <NRf> specifies the how long waveform points are displayed on the screen. Range: 0 s to 1000 s (the default value is 10 ms).

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

TRACE5:AVTIME:DOT:PERSISTENT:VARIABLE 20 specifies that the waveform points are displayed on the screen for a period of 20 before they disappear.

TRACe<x>:AVTime[:MEASview<y>]:FREeze

Determines whether or not to freeze the display of the specified trace in the Amplitude versus Time view.

The parameter <x> = 1 to 7; All traces are valid.

Conditions

Measurement views: Amplitude versus Time

Group

Trace commands

Syntax

```
TRACe<x>:AVTime[:MEASview<y>]:FREeze { OFF | ON | 0 | 1 }
TRACe<x>:AVTime[:MEASview<y>]:FREeze?
```

Arguments

OFF or 0 updates the display of the specified trace normally.

ON or 1 stops updating the display of the specified trace.

The parameter <x> = 1 to 7.

- Trace 4: Math trace
- Trace 5: Bitmap trace
- Trace 6: Ogram
- Trace 7: Ogram Line

The parameter <y>= 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

TRACE1:AVTIME:MEASVIEW1:FREEZE ON freezes the display for Trace 1 in the Amplitude versus Time Measurement View 1.

TRACe<x>:AVTime[:MEASview<y>]:FUNCtion

Sets or queries the function for the specified trace in the Amplitude versus Time view.

The parameter <x> = 1 to 3; Trace 4 (math trace) is invalid.

Conditions

Measurement views: Amplitude versus Time

Group

Trace commands

Syntax

```
TRACe<x>:AVTime[:MEASview<y>]:FUNCtion { NORMal | AVERage | MAXHold | MINHold }
TRACe<x>:AVTime[:MEASview<y>]:FUNCtion?
```

Arguments

NORMal selects the normal display.

AVERage selects the Average display that indicates the average amplitude at each time point.

MAXHold selects the Max Hold display that indicates the maximum amplitude at each time point.

MINHold selects the Min Hold display that indicates the minimum amplitude at each time point.

The parameter <x> = 1 to 3.

The parameter <y>= 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

TRACE1:AVTIME:MEASVIEW1:FUNCTION MAXHold selects Max Hold for Trace 1 in the Amplitude versus Time Measurement View 1.

TRACe<x>:AVTime[:MEASview<y>]:LEFToperand

Sets or queries the left operand for the math trace (Trace 4) in the Amplitude versus Time view.

The parameter <x> = 4; Only Trace 4 (math trace) is valid.

Conditions

Measurement views: Amplitude versus Time

Group

Trace commands

Syntax

```
TRACe<x>:AVTime[:MEASview<y>]:LEFToperand { TRACE1 | TRACE2 | TRACE3 }
TRACe<x>:AVTime[:MEASview<y>]:LEFToperand?
```

Related Commands

TRACe<x>:AVTime[:MEASview<y>]:RIGHtoperand (on page 2629)

Arguments

`TRACe1` selects Trace 1 as the left operand for the math trace.

`TRACe2` selects Trace 2 as the left operand for the math trace.

`TRACe3` selects Trace 3 as the left operand for the math trace.

The parameter <x> = 4; Only Trace 4 (math trace) is valid.

The parameter <y>= 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`TRACE2:AVTIME:MEASVIEW1:LEFTOPERAND` selects Trace 2 as the left operand for the math trace for Amplitude versus Time Measurement View 1.

TRACe<x>:AVTime[:MEASview<y>]:RIGHtoperand

Sets or queries the right operand for the math trace (Trace 4) in the Amplitude versus Time view.

The parameter <x> = 4; Only Trace 4 (math trace) is valid.

Conditions

Measurement views: Amplitude versus Time

Group

Trace commands

Syntax

```
TRACe<x>:AVTime[:MEASview<y>]:RIGHtoperand { TRACE1 | TRACE2 | TRACE3 }
TRACe<x>:AVTime[:MEASview<y>]:RIGHtoperand?
```

Related Commands

TRACe<x>:AVTime[:MEASview<y>]:LEFToperand (on page 2628)

Arguments

`TRACe1` selects Trace 1 as the right operand for the math trace.

`TRACe2` selects Trace 2 as the right operand for the math trace.

`TRACe3` selects Trace 3 as the right operand for the math trace.

The parameter <x> = 4; Only Trace 4 (math trace) is valid.

The parameter <y>= 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`TRACE1:AVTIME:MEASVIEW1:RIGHTOPERAND` selects Trace 1 as the right operand for the math trace on Amplitude versus Time Measurement View 1.

TRACe<x>:AVTime[:MEASview<y>]:SELect

Sets or queries the trace whose measurement results are being displayed in the readout on the top and bottom of the view.

The parameter <x> = 1 to 7; All traces are valid.

Conditions

Measurement views: Amplitude versus Time

Group

Trace commands

Syntax

```
TRACe<x>:AVTime[:MEASview<y>]:SELect {ON | 1}
TRACe<x>:AVTime[:MEASview<y>]:SELect?
```

Arguments

`ON` or `1` selects the specified trace

The parameter <x> = 1 to 7.

- Trace5: Bitmap Trace
- Trace6: Ogram
- Trace7: Ogram Line

The parameter <y>= 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Returns

`{ 0 | 1 }`

- 0 indicates that the results are not being displayed in the readout for the trace.
- 1 indicates that the results are being displayed in the readout for the trace.

Examples

`TRACE1:AVTIME:MEASVIEW1:SELECT 1` selects Trace 1 to display the measurement results in the readout forAmplitude versus Time Measurement View 1.

`TRACE1:AVTIME:MEASVIEW1:SELECT?` might return `1`, indicating that the results are being displayed in the readout for Trace 1 on Amplitude versus Time Measurement View 1.

TRACe<x>:AVTime[:MEASview<y>]:SELect:LINe

Sets or queries the line number to send to the Ampliude vs Time ogram display.

Conditions

Applicable measurement view must be active.

Group

Trace commands

Syntax

`TRACe<x>:AVTime[:MEASview<y>]:SELect:LINe <number>`

`TRACe<x>:AVTime[:MEASview<y>]:SELect:LINe?`

Related Commands

TRACe<x>:AVTime[:MEASview<y>] (on page 2621)

Arguments

<y> = 1 to 8 represents measurement index based on number of sources of connected instrument. where MEASview keyword is optional. By default y is 1.

<x> = 6 for ogram trace.

`<number>::=<NR1>` specifies the line number to send to the display.

Range: 0 to the maximum line number of the ogram displayed on screen.

Returns

`<NRf>` specifies the line number to send to the spectrum display.

Examples

`TRACe6:AVTime:MEASview1:SELect:LINe 75` selects Line #75 in the ogram to send to the time plot display.

TRACe<x>:BLUEtooth:CONSte

Sets or queries the specified trace to display or hide in the Bluetooth Constellation display.

The parameter <x> is trace 1 or 2.

Conditions

Measurement views: Bluetooth Constellation

Group

Trace commands

Syntax

```
TRACe<x>:BLUEtooth:CONSte { OFF | ON | 0 | 1 }
TRACe<x>:BLUEtooth:CONSte?
```

Arguments

`OFF` or 0 hides the selected trace.

`ON` or 1 displays the selected trace.

Returns

0 indicates the specified trace is not shown in the Bluetooth Constellation display.

1 indicates the specified trace is shown in the Bluetooth Constellation display.

Examples

TRACe1:BLUEtooth:CONSte ON specifies that trace 1 appears in the Bluetooth Constellation display.

TRACe<x>:BLUEtooth:CONSte:FREeze

Sets or queries the specified trace to freeze or not to freeze in the Bluetooth Constellation display.

The parameter <x> is trace 1 or 2.

Conditions

Measurement views: Bluetooth Constellation

Group

Trace commands

Syntax

```
TRACe<x>:BLUEtooth:CONSte:FREeze {OFF | ON | 0 | 1 }
TRACe<x>:BLUEtooth:CONSte:FREeze?
```

Arguments

OFF or 0 updates the display of the selected trace normally.

ON or 1 stops updating the display of the selected trace.

Returns

0 indicates the specified trace is being updated in the Bluetooth Constellation display.

1 indicates the specified trace is not being updated in the Bluetooth Constellation display.

Examples

TRACe1:BLUEtooth:CONSte:FREeze ON freezes the display for trace 1 in the Bluetooth Constellation display.

TRACe<x>:BLUEtooth:CONSte:MODE

Sets or queries the Bluetooth Constellation trace display mode. The parameter <x> is trace 1 or 2.

Conditions

Measurement views: Bluetooth Constellation

Group

Trace commands

Syntax

```
TRACe<x>:BLUEtooth:CONSte:MODE { VECT | SYMB| LIN }
TRACe<x>:BLUEtooth:CONSte:MODE?
```

Arguments

VECT connects adjacent symbol pints with the signal locus.

SYMB displays individual symbol points. LIN displays linear symbol points.

Examples

TRACe1:BLUEtooth:XONSte:MODE VECT shows the Bluetooth Constellation display is connecting adjacent symbol points with the signal locus.

TRACe<x>:BLUEtooth:CONSte:SELect

Sets or queries the active trace in the Bluetooth Constellation diagram. The parameter <x> is trace 1 or 2.

Conditions

Measurement views: Bluetooth Constellation

Group

Trace commands

Syntax

```
TRACe<x>:BLUEtooth:CONSte:SELect <active>
TRACe<x>:BLUEtooth:CONSte:SELect?
```

Arguments

```
<active> ::= {0 | 1}
```

Where:

`0`, the specified trace is not the selected trace.

`1`, the specified trace is the selected trace.

Examples

`TRACe1:BLUEtooth:CONSte:SELect` specifies trace 1 as the selected trace.

TRACe<x>:BLUEtooth:EDIagram:ENABle:I

Determines whether to show or hide the I trace in the Bluetooth eye diagram.

Conditions

Measurement views: Bluetooth Eye diagram

Group

Trace commands

Syntax

```
TRACe<x>:BLUEtooth:EDIagram:ENABle:I { OFF | ON | 0 | 1 }
TRACe<x>:BLUEtooth:EDIagram:ENABle:I?
```

Arguments

`OFF` or 0 hides the I trace.

`ON` or 1 shows the I trace.

Examples

`TRACEX:BLUETOOTH:EDIAGRAM:ENABLE:I ON` shows the I trace in the eye diagram.

TRACe<x>:BLUEtooth:EDIagram:ENABle:Q

Determines whether to show or hide the Q trace in the Bluetooth Eye diagram.

Conditions

Measurement views: Bluetooth Eye diagram

Group

Trace commands

Syntax

```
TRACe<x>:BLUEtooth:EDIagram:ENABle:Q { OFF | ON | 0 | 1 }
TRACe<x>:BLUEtooth:EDIagram:ENABle:Q?
```

Arguments

`OFF` or 0 hides the Q trace.

`ON` or 1 shows the Q trace.

Examples

`TRACEX:BLUETOOTH:EDIAGRAM:ENABLE:Q ON` shows the Q trace in the eye diagram.

TRACe<x>:BLUEtooth:EDIagram:SELect:I

Selects the I trace in the Bluetooth eye diagram. The query version of this command returns whether the I trace is selected or not.

Conditions

Measurement views: Bluetooth Eye diagram

Group

Trace commands

Syntax

```
TRACe<x>:BLUEtooth:EDIagram:SELect:I { 0 | 1 }
TRACe<x>:BLUEtooth:EDIagram:SELect:I?
```

Arguments

None

Returns

0 indicates that the I trace is deselected.

1 indicates that the I trace is selected.

Examples

`TRACEX:BLUETOOTH:EDIAGRAM:SELECT:I` selects the I trace in the eye diagram.

TRACe<x>:BLUEtooth:EDIagram:SELect:Q

Selects the Q trace in the Bluetooth eye diagram. The query version of this command returns whether the I trace is selected or not.

Conditions

Measurement views: Bluetooth Eye diagram

Group

Trace commands

Syntax

```
TRACe<x>:BLUEtooth:EDIagram:SELect:Q { 0 | 1 }
TRACe<x>:BLUEtooth:EDIagram:SELect:Q?
```

Arguments

None

Returns

0 indicates that the Q trace is deselected.

1 indicates that the Q trace is selected.

Examples

`TRACEX:BLUETOOTH:EDIAGRAM:SELECT:Q` selects the Q trace in the eye diagram.

TRACe<x>:BLUEtooth:FDVTime:MODE

Sets or queries whether to display the Frequency Deviation vs. Time trace as vectors or symbols (points).

Conditions

Measurement views: Frequency Deviation vs. Time

Group

Trace commands

Syntax

```
TRACe<x>:BLUEtooth:FDVTime:MODE { VECTors | SYMBols }
TRACe<x>:BLUEtooth:FDVTime:MODE?
```

Arguments

`VECTors` connects adjacent symbol points with lines.

`SYMBols` displays individual symbol points.

Examples

`TRACe:BLUEtooth:FDVTime:MODE VECT` sets the display to connect points in the display with lines.

TRACe<x>:CCDF:FREeze

Determines whether or not to freeze the display of the specified trace (Trace 1 or 2) in the CCDF view.

The parameter <x> = 1 or 2; Trace 3 (Gaussian curve) is invalid.

Conditions

Measurement views: CCDF

Group

Trace commands

Syntax

```
TRACe<x>:CCDF:FREeze { OFF | ON | 0 | 1 }
```

TRACe<x>:CCDF:FREeze?

Arguments

OFF or 0 updates the display of the specified trace normally.

ON or 1 stops updating the display of the specified trace.

Examples

TRACE1:CCDF:FREEZE ON freezes the display for Trace 1.

TRACe<x>:CCDF:SELect

Sets or queries the trace whose measurement results are being displayed in the readout on the top of the view. The selected trace is indicated by the measurement pointer (pink triangle) on the waveform.

The parameter <x> = 1 to 3; All traces are valid.

Conditions

Measurement views: CCDF

Group

Trace commands

Syntax

TRACe<x>:CCDF:SELect
TRACe<x>:CCDF:SELect?

Arguments

None

Returns

{ 0 | 1 }

0 indicates that the results are not being displayed in the readout for the trace.

1 indicates that the results are being displayed in the readout for the trace.

Examples

TRACE1:CCDF:SELECT selects Trace 1 to display the measurement results in the readout.

TRACE1:CCDF:SELECT? might return 1, indicating that the results are being displayed in the readout for Trace 1.

TRACe<x>:CCDF:SHOW

Determines whether to show or hide the specified trace in the CCDF view. The parameter <x> = 1 to 3; All traces are valid.

Conditions

Measurement views: CCDF

Group

Trace commands

Syntax

```
TRACe<x>:CCDF:SHOW { OFF | ON | 0 | 1 }
TRACe<x>:CCDF:SHOW?
```

Arguments

OFF or 0 hides the specified trace.

ON or 1 shows the specified trace.

Examples

TRACE1:CCDF:SHOW ON shows Trace 1 in the CCDF view.

TRACe<x>:CCDF:X

Sets or queries the horizontal position of the measurement pointer (pink triangle) to measure the CCDF. Use the TRACe<x>:CCDF:Y? query to read the value.

The parameter <x> = 1 to 3.

NOTE: Use the TRACe<x>:CCDF:SHOW command to show the specified trace and the TRACe<x>:CCDF:SELect command to select the trace before running the TRACe<x>:CCDF:X command.

Conditions

Measurement views: CCDF

Group

Trace commands

Syntax

```
TRACe<x>:CCDF:X <value>
TRACe<x>:CCDF:X?
```

Arguments

<value>::<NRf> specifies the horizontal position of the measurement pointer.

Range: 0 to 20 dB.

Examples

TRACE1:CCDF:X 5 puts the measurement pointer at 5 dB on Trace 1.

TRACe<x>:CCDF:Y? (Query Only)

Queries the vertical position (CCDF value) of the measurement pointer (displayed as a pink triangle). Use the TRACe<x>:CCDF:X command to set the horizontal position of the pointer.

The parameter <x> = 1 to 3.

NOTE: Use the TRACe<x>:CCDF:SHOW command to show the specified trace and the TRACe<x>:CCDF:SELect command to select the trace before running the TRACe<x>:CCDF:Y? query.

Conditions

Measurement views: CCDF

Group

Trace commands

Syntax

`TRACe<x>:CCDF:Y?`

Arguments

None

Returns

`<value>::<NRf>` is the vertical position (CCDF) of the measurement pointer.

Range: 0 to 100%.

The value of 99.0999953003E+36 is returned if the trace is not available.

Examples

`TRACE1:CCDF:Y?` might return 14.72, indicating the CCDF is 14.72% at the measurement pointer on Trace 1.

TRACe<x>:CONSte[:MEASview<y>]

Specifies whether or not to show the specified trace (Trace 1 or 2) in the Constellation display.

The parameter <x> = 1 or 2.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Constellation

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Constellation).

Group

Trace commands

Syntax

```
TRACe<x>:CONSte[:MEASview<y>] {OFF | ON | 0 | 1 }
TRACe<x>:CONSte[:MEASview<y>]?
```

Arguments

`OFF` or 0 hides the selected trace.

`ON` or 1 displays the selected trace.

Returns

`{ 0 }` indicates the specified trace is not shown in the Constellation display.

`{ 1 }` indicates the specified trace is shown in the Constellation display.

Examples

`TRACe1:CONSTe:MEASview2 2` specifies that Trace 2 be shown in the Constellation display for measview 2 of constellation.

TRACe<x>:CONSte[:MEASview<y>]:FREeze

Specifies whether or not to freeze (halt updates) to the display of the specified trace (Trace 1 or 2) in the Constellation display.

The parameter <x> = 1 or 2.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Constellation

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Constellation) .

Group

Trace commands

Syntax

```
TRACe<x>:CONSte[:MEASview<y>]:FREeze {OFF | ON | 0 | 1 }
TRACe<x>:CONSte[:MEASview<y>]:FREeze?
```

Arguments

`OFF` or 0 updates the display of the specified trace normally.

`ON` or 1 stops updating the display of the specified trace.

Returns

`{ 0 }` indicates the specified trace is being updated.

`{ 1 }` indicates the specified trace is not being updated.

Examples

`TRACe1:CONSTe:MEASview2:FREeze ON` freezes the display for Trace 1 for measview 2 of constellation.

TRACe<x>:CONSte[:MEASview<y>]:MODE

Sets or queries how to display the constellation trace.

The parameter <x> = 1 or 2.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Constellation

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Constellation).

Group

Trace commands

Syntax

```
TRACe<x>:CONSte[:MEASview<y>]:MODE { VECTors | SYMBols }
TRACe<x>:CONSte[:MEASview<y>]:MODE?
```

Arguments

`VECTors` connects adjacent symbol points with the signal locus.

`SYMBols` displays individual symbol points.

Examples

`TRACe<X>:CONSTe:MEASview2:MODE VECTors` shows the constellation connecting adjacent symbol points with the signal locus for measview 2 of constellation.

TRACe<x>:CONSte[:MEASview<y>]:Q:OFFSet

Sets the Q offset in the Constellation display to Use shared Pref, Remove Q offset, or Include Q offset.

The parameter <x> = 1 or 2.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Constellation

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Constellation).

Group

Trace commands

Syntax

```
TRACe<x>:CONSte[:MEASview<y>]:Q:OFFSet { SHARed | RQOFset | IQOFset }
TRACe<x>:CONSte[:MEASview<y>]:Q:OFFSet?
```

Arguments

`SHARed` means that the Q offset applied to the selected trace will be the same as the Remove Q offset setting located on the Prefs tab (which applies to all GP Digital Modulation displays).

Remove Q offset removes the Q offset, possibly overriding the Remove Q offset setting on the Prefs tab. Changing the Constellation display Q Offset value with this command does not affect the Q offset setting on other displays.

Include Q offset includes the Q offset, possibly overriding the Remove Q offset setting on the Prefs tab. Changing the Constellation display Q Offset value with this command does not affect the Q offset setting on other displays.

Returns

`{ SHAR }` indicates that Q Offset is set to Use Shared Pref.

`{ RQOF }` indicates that Q Offset is set to Remove Q Offset.

`{ IQOF }` indicates that Q Offset is set to Include Q Offset.

Examples

`TRACe1:CONSte:MEASview2:Q:OFFSet SHAR` sets the Constellation display Q Offset to Use Shared Pref for measview 2 of constellation.

TRACe<x>:CONSte[:MEASview<y>]:SELect (No query form)

Sets or queries the trace to which the Show, Freeze, Content, and Q Offset settings apply.

The parameter <x> = 1 or 2.

The parameter <y> = 1 to number of instances of a measurement created or number of channels available in an instrument (Example: 1 to 8 channels for MSO58).

Conditions

Measurement views: Constellation

MEASview keyword is optional and applicable only for measurements that supports multi-instance creation of measurement (Example: GPDemod Constellation).

Group

Trace commands

Syntax

```
TRACe<x>:CONSte[:MEASview<y>]:SELect
TRACe<x>:CONSte[:MEASview<y>]:SELect?
```

Arguments

None

Returns

`{ 0 }` indicates the specified trace is not the selected trace.

`{ 1 }` indicates the specified trace is the selected trace.

Examples

`TRACe1:CONSte:MEASview2:SELect` specifies Trace 1 as the selected trace for measview 2 of constellation.

TRACe<x>:DPX

Determines whether or not to show the specified trace in the DPX, Spectrum view.

The parameter <x> = 1 to 7; All traces are valid. Trace 6 is for the DPXogram trace and Trace 7 is for the Ogram line.

Conditions

Measurement views: DPX, Spectrum

Group

Trace commands

Syntax

```
TRACe<x>:DPX { OFF | ON | 0 | 1 }
TRACe<x>:DPX?
```

Arguments

`OFF` or 0 hides the specified trace in the DPX, Spectrum view.

`ON` or 1 shows the specified trace in the DPX, Spectrum view.

Examples

`TRACE1:DPX ON` shows Trace 1 (the maximum trace) in the DPX, Spectrum view.

TRACe<x>:DPX:AVERage:COUNt

Sets or queries the number of traces to combine for averaging in the DPX, Spectrum view.

The parameter <x> = 3; Only Trace 3 (average trace) is valid.

Conditions

Measurement views: DPX, Spectrum

Group

Trace commands

Syntax

```
TRACe<x>:DPX:AVERage:COUNt <number>
TRACe<x>:DPX:AVERage:COUNt?
```

Arguments

`<number>::<NR1>` specifies the number of traces to combine for averaging.

Range: 1 to 10000.

Examples

`TRACE3:DPX:AVERAGE:COUNT 32` sets the average count to 32.

TRACe<x>:DPX:COLor:CURVe

Sets or queries how colors are mapped to the signal density in the DPX, Spectrum bitmap display. The mapping can be linear (Curve = 1), or it can be set to concentrate the resolution on the lower level of the range (Curve > 1) or the mapping can be set to show the best resolution on the upper range of density or hit count (Curve = 0.1 – 0.99). This command is only valid for the Bitmap display.

The parameter <x> = 5; only Trace5 is valid.

Conditions

Measurement views: DPX, Spectrum

Group

Trace commands

Syntax

```
TRACe<x>:DPX:COLor:CURVe <value>
TRACe<x>:DPX:COLor:CURVe?
```

Arguments

`<value>::<NRf>` specifies how colors are mapped to the signal density. Range:

Examples

`TRACe<x>:DPX:COLor:CURVe 1.5` concentrates the resolution on the lower level of the range on the Bitmap display.

TRACe<x>:DPX:COLor:INTensity

Sets or queries the color intensity in the DPX, Spectrum view. The value is common to all traces.

The parameter <x> = 1 to 5; All traces are valid.

Conditions

Measurement views: DPX, Spectrum

Group

Trace commands

Syntax

```
TRACe<x>:DPX:COLor:INTensity <value>
TRACe<x>:DPX:COLor:INTensity?
```

Arguments

`<value>::<NRf>` specifies color intensity. Range: 1 to 100%.

Examples

`TRACE1:DPX:COLOR:INTENSITY 30` sets the color intensity to 30%.

TRACe<x>:DPX:COLor:SCALe:AUTO (No Query Form)

Automatically adjusts the Max and Min color settings to display the broadest range of colors in the DPX, Spectrum bitmap display. This command is only valid for the Bitmap display.

The parameter <x> = 5; only Trace5 is valid.

Conditions

Measurement views: DPX, Spectrum

Group

Trace commands

Syntax

```
TRACe<x>:DPX:COLor:SCALe:AUTO
```

Arguments

`<x>::=5` specifies the auto-color setting for the Bitmap trace.

Examples

`TRACe5:DPX:COLor:SCALe:AUTO` autosets the Bitmap display color scheme.

TRACe<x>:DPX:DETection

Sets or queries the detection type (POSitive, NEGative, AVERage) for traces 1, 2 and 3 in the DPX, Spectrum view.

Trace 4 (math trace) and Trace 5 (bitmap trace) are invalid choices. Trace 6 is only valid for DPXogram.

Conditions

Measurement views: DPX, Spectrum

Group

Trace commands

Syntax

```
TRACe<x>:DPX:DETection { AVERage | NEGative | POSitive }
```

TRACe<x>:DPX:DETection?

Arguments

The following table shows the trace function and display. For the average trace, use the TRACe<x>:DPX:AVERage:COUNt command to set the average count.

Examples

TRACE1:DPX:DETection AVERage enables analysis to look for AVERage on Trace 1.

TRACe<x>:DPX:DGRAM:SELect:LINE

Sets or queries a line from the DPXogram display to send to the DPX, Spectrum display.

Trace 6 is the DPXogram trace. No other traces are allowed.

Conditions

Measurement views: DPX, Spectrum

Group

Trace commands

Syntax

TRACe<x>:DPX:DGRAM:SELect:LINE <number>
TRACe<x>:DPX:DGRAM:SELect:LINE?

Arguments

<number> ::= <NR1> specifies the line number in the DPXogram display.

Examples

TRACE6:DPX:DGRAM:SELECT:LINE 75 selects line #75 in the DPXogram display to send to the DPX, Spectrum display.

TRACe<x>:DPX:DOT:PERSistent

Determines whether to enable or disable the dot persistence for the bitmap trace (Trace 5) in the DPX, Spectrum view.

The parameter <x> = 5; Only Trace 5 (bitmap trace) is valid.

Conditions

Measurement views: DPX, Spectrum

Group

Trace commands

Syntax

TRACe<x>:DPX:DOT:PERSistent { OFF | ON | 0 | 1 }
TRACe<x>:DPX:DOT:PERSistent?

Arguments

OFF or 0 disables the dot persistence.

ON or 1 enables the dot persistence.

Examples

`TRACE5:DPX:DOT:PERSISTENT ON` enables the dot persistence in the DPX, Spectrum view.

TRACe<x>:DPX:DOT:PERSistent:TYPE

Sets or queries the persistence type for the bitmap trace (Trace 5) in the DPX, Spectrum view.

The parameter <x> = 5; Only Trace 5 (bitmap trace) is valid.

Conditions

Measurement views: DPX, Spectrum

Group

Trace commands

Syntax

```
TRACe<x>:DPX:DOT:PERSistent:TYPE { VARiable | INFinite }
TRACe<x>:DPX:DOT:PERSistent:TYPE?
```

Arguments

`VARiable` selects the variable persistence display which leaves acquired data points on the display for a period of time specified by the `TRACe<x>:DPX:DOT:PERSistent:VARiable` command.

`INFinite` selects the infinite persistence display which accumulates data points on the display indefinitely.

Examples

`TRACE5:DPX:DOT:PERSISTENT:TYPE VARiable` selects the variable persistence display.

TRACe<x>:DPX:DOT:PERSistent:VARiable

Sets or queries how long data points are displayed. This command is effective when `TRACe<x>:DPX:DOT:PERSistent:TYPE` is set to `VARiable`. This affects the display only.

The parameter <x> = 5; Only Trace 5 (bitmap trace) is valid.

Conditions

Measurement views: DPX, Spectrum

Group

Trace commands

Syntax

```
TRACe<x>:DPX:DOT:PERSistent:VARiable <number>
TRACe<x>:DPX:DOT:PERSistent:VARiable?
```

Arguments

`<number>::<NR1>` specifies the how long waveform points are displayed on the screen. Range: 100 ms to 60 s (the default value is 1.0 s).

Examples

`TRACE5:DPX:DOT:PERSISTENT:VARIABLE 20` specifies that the waveform points are displayed on the screen for a period of 20 before they disappear.

TRACe<x>:DPX:FREeze

Determines whether or not to freeze the display of the specified trace in the DPX, Spectrum view.

The parameter <x> = 1 to 6; All traces are valid. Trace 6 is for the DPXogram trace and Trace 7 is for the Ogram line.

Conditions

Measurement views: DPX, Spectrum

Group

Trace commands

Syntax

```
TRACe<x>:DPX:FREeze { OFF | ON | 0 | 1 }
TRACe<x>:DPX:FREeze?
```

Arguments

`OFF` or 0 updates the display of the specified trace normally.

`ON` or 1 stops updating the display of the specified trace.

Examples

`TRACE1:DPX:FREEZE ON` freezes the display for the +peak trace.

TRACe<x>:DPX:FUNCtion

Sets or queries the trace function for the +Peak, -Peak, or Average trace (Trace 1, 2, or 3, respectively) in the DPX, Spectrum view.

The parameter <x> = 1 to 3; Trace 4 (math trace) and Trace 5 (bitmap trace) are invalid.

Conditions

Measurement views: DPX, Spectrum

Group

Trace commands

Syntax

```
TRACe<x>:DPX:FUNCtion { NORMal | HOLD | AVERage }
TRACe<x>:DPX:FUNCtion?
```

Arguments

The following table shows the trace function and display. For the average trace, use the `TRACe<x>:DPX:AVERage:COUNt` command to set the average count.

Function	Trace1 (+Peak trace)	Trace1 (+Peak trace)	Trace3 (Average trace)
NORMal	Normal spectrum (Detection: +Peak)	Normal spectrum (Detection: -Peak)	Normal spectrum (Detection: Average)
HOLD	Max-hold spectrum	Min-hold spectrum	NA
AVERage	NA	NA	Average spectrum

Examples

`TRACE1:DPX:FUNCTION HOLD` selects the max hold waveform for Trace 1.

TRACe<x>:DPX:LEFToperand

Sets or queries the left operand for the math trace (Trace 4) in the DPX, Spectrum view.

The parameter <x> = 4; Only Trace 4 (math trace) is valid.

Conditions

Measurement views: DPX, Spectrum

Group

Trace commands

Syntax

```
TRACe<x>:DPX:LEFToperand { TRACe1 | TRACe2 | TRACe3 }
TRACe<x>:DPX:LEFToperand?
```

Related Commands

TRACe<x>:DPX:RIGHtoperand (on page 2647)

Arguments

`TRACE1` selects Trace 1 as the left operand for the math trace.

`TRACE2` selects Trace 2 as the left operand for the math trace.

`TRACE3` selects Trace 3 as the left operand for the math trace.

Examples

`TRACE4:DPX:LEFTOPERAND TRACE2` selects Trace 2 as the left operand for the math trace.

TRACe<x>:DPX:RIGHtoperand

Sets or queries the right operand for the math trace (Trace 4) in the DPX, Spectrum view.

The parameter <x> = 4; Only Trace 4 (math trace) is valid.

Conditions

Measurement views: DPX, Spectrum

Group

Trace commands

Syntax

```
TRACe<x>:DPX:RIGHtoperand { TRACE1 | TRACE2 | TRACE3 }
TRACe<x>:DPX:RIGHtoperand?
```

Related Commands

TRACe<x>:DPX:LEFToperand (on page 2647)

Arguments

`TRACE1` selects Trace 1 as the right operand for the math trace.

TRACE2 selects Trace 2 as the right operand for the math trace.

TRACE3 selects Trace 3 as the right operand for the math trace.

Examples

TRACE4:DPX:RIGHTOPERAND TRACE1 selects Trace 1 as the right operand for the math trace.

TRACe<x>:DPX:SELect

Sets or queries the trace to display the readout at the upper left of the DPX, Spectrum view.

The parameter <x> = 1 to 7; All traces are valid. Trace 6 is for the DPXogram trace and Trace 7 is for the Ogram line.

Conditions

Measurement views: DPX, Spectrum

Group

Trace commands

Syntax

```
TRACe<x>:DPX:SELect
TRACe<x>:DPX:SELect?
```

Arguments

None

Returns

{ 0 | 1 }

- 0 indicates that the readout is not being displayed for the specified trace.
- 1 indicates that the readout is being displayed for the specified trace.

Examples

TRACE1:DPX:SELECT selects Trace 1 (+Peak trace) to display the readout.

TRACE1:DPX:SELECT? might return 1, indicating that the readout is being displayed for Trace 1.

TRACe<x>:EMC:RECAll:ENABle

Sets or queries that selected recalled trace is enabled (displayed) or disabled in EMC View.

<x> = <NR1> either 1,2,3,4,5 (trace number)

Conditions

Measurement view: EMC - EMI Display

Group

Trace commands

Syntax

```
TRACe<x>:EMC:RECAll:ENABle {0 | OFF | 1 | ON}
TRACe<x>:EMC:RECAll:ENABle?
```

Arguments

0 or OFF indicates recalled trace is disabled 1 or ON indicates recalled trace is enabled

Returns

<Boolean> 0,1

Examples

TRACe4:EMC:RECAll:ENABle might return 1, indicating that the recalled trace 4 is enabled (displayed).

TRACe<x>:EMC:RECAll:SELect

Sets or queries the selection of specified recalled trace in live trace of EMC View.

<x> = <NR1> either 1,2,3,4,5 (trace number)

Conditions

Measurement view: EMC - EMI Display

Group

Trace commands

Syntax

```
TRACe<x>:EMC:RECAll:SELect
TRACe<x>:EMC:RECAll:SELect?
```

Arguments

None

Returns

<Boolean> 0,1

Examples

TRACe3:EMC:RECAll:SELect might return 1, indicating that recalled trace 3 is selected.

TRACe<x>:EMC:SCAN:COUNt

Sets or queries the count value of specified scan trace in EMC View.

<x> = <NR1> 1 or 2 (trace number)

Conditions

Measurement view: EMC - EMI Display

Group

Trace commands

Syntax

```
TRACe<x>:EMC:SCAN:COUNt <value>
TRACe<x>:EMC:SCAN:COUNt?
```

Arguments

<value> = <NR1>

Returns

<NR1>

Examples

TRACe1:EMC:SCAN:COUNt might return 12, indicating the count value for trace 1 is 12.

TRACe<x>:EMC:SCAN:COUNt:RESet (No Query Form)

Resets the count value of the specified live scan trace in EMC view.

<x> = <NR1> 1 or 2 (trace number)

Conditions

Measurement view: EMC - EMI Display

Group

Mass Memory commands

Syntax

TRACe<x>:EMC:SCAN:COUNt:RESet

Arguments

None

Returns

None

Examples

TRACe2:EMC:SCAN:COUNt:RESet resets the count value of live scan trace 2.

TRACe<x>:EMC:SCAN:ENABle

Sets or queries that selected scan trace is enabled (displayed) or disabled in EMC View.

<x> = <NR1> 1 or 2 (trace number)

Conditions

Measurement view: EMC - EMI Display

Group

Trace commands

Syntax

```
TRACe<x>:EMC:SCAN:ENABle { 0 | OFF | 1 | ON}
TRACe<x>:EMC:SCAN:ENABle?
```

Arguments

0 or OFF indicates the scan trace is disabled. 1 or ON indicates the scan trace is enabled.

Returns

<Boolean> 0,1

Examples

TRACe1:EMC:SCAN:ENABle? might return 1, indicating that the scan trace 1 is enabled (displayed).

TRACe<x>:EMC:SCAN:FUNCtion

Sets or queries the scan function for the selected scan trace in EMC View.

<x> = <NR1>1 or 2 Trace number

Conditions

Measurement view: EMC - EMI Display

Group

Trace commands

Syntax

```
TRACe<x>:EMC:SCAN:FUNCtion {NONE | MAXH | AVER | AVGL}
TRACe<x>:EMC:SCAN:FUNCtion?
```

Arguments

NONE = Normal is the function selected MAXH = Max Hold is the function selected AVER = Avg(VRMS) is the function selected

AVGL = Avg (of logs) is the function selected

Returns

<String> See Arguments

Examples

TRACe1:EMC:SCAN:ENABle? might returns MAXH, indicating that the scan function selected for trace 1 is Max Hold.

TRACe<x>:EMC:SCAN:SELect

Sets or queries the selection of scan trace in live trace of EMC View.

<x> = <NR1>1 or 2 (trace number)

Conditions

Measurement view: EMC - EMI Display

Group

Trace commands

Syntax

```
TRACe<x>:EMC:SCAN:SELect
TRACe<x>:EMC:SCAN:SELect?
```

Arguments

None

Returns

<Boolean> 0,1

Examples

`TRACe1:EMC:SCAN:SELect` might return `1`, indicating that scan trace 1 is selected.

TRACe<x>:EMC:SPOT:ENABle

Sets or queries that selected spot trace is enabled (displayed) or disabled in EMC View.

<x> = <NR1> either 1 , 2 or 3 (trace number)

Conditions

Measurement view: EMC - EMI Display

Group

Trace commands

Syntax

```
TRACe<x>:EMC:SPOT:ENABle {0 | OFF | 1 | ON}
TRACe<x>:EMC:SPOT:ENABle?
```

Arguments

`0` or `OFF` indicates spot trace is disabled 1 or ON indicates spot trace is enabled

Returns

`<Boolean>` 0,1

Examples

`TRACe2:EMC:SPOT:ENABle` might return 1, indicating that the spot trace 2 is enabled (displayed).

TRACe<x>:EMC:SPOT:SELect

Sets or queries the selection of spot trace in live trace of EMC View.

<x> = <NR1> either 1 , 2 or 3 (trace number)

Conditions

Measurement view: EMC - EMI Display

Group

Trace commands

Syntax

```
TRACe<x>:EMC:SPOT:SELect
TRACe<x>:EMC:SPOT:SELect?
```

Arguments

None

Returns

`<Boolean>` 0,1

Examples

TRACe1:EMC:SPOT:SELect might return 1, indicating that spot trace 1 is selected.

TRACe<x>:{FSETtling|PSETtling}:FREeze

Determines whether to freeze the specified trace in the Settling Time measurement display.

The parameter <x> = 1 or 2.

Conditions

Measurement views: Frequency and Phase Settling Time

Group

Trace commands

Syntax

```
TRACe<x>:{FSETtling|PSETtling}:FREeze { OFF | ON | 0 | 1 }
TRACe<x>:{FSETtling|PSETtling}:FREeze?
```

Arguments

OFF or 0 updates the trace display normally.

ON or 1 stops updating trace display.

Examples

TRACE1:FSETTLING:FREEZE ON stops updating the Trace 1 display.

TRACe<x>:{FSETtling|PSETtling}:SELect

Selects the trace in the settling time measurement. The query returns the currently selected trace.

The parameter <x> = 1 or 2.

Conditions

Measurement views: Frequency and Phase Settling Time

Group

Trace commands

Syntax

```
TRACe<x>:{FSETtling|PSETtling}:SELect
TRACe<x>:{FSETtling|PSETtling}:SELect?
```

Arguments

None

Returns

0 (not selected) or 1 (selected).

Examples

TRACE2:FSETTLING:SELECT selects Trace 2.

TRACe<x>:{FSETtling|PSETtling}:SHOW

Determines whether to show or hide the specified trace in the settling time view. The parameter <x> = 1 or 2.

Conditions

Measurement views: Frequency and Phase Settling Time

Group

Trace commands

Syntax

```
TRACe<x>:{FSETtling|PSETtling}:SHOW { OFF | ON | 0 | 1 }
TRACe<x>:{FSETtling|PSETtling}:SHOW?
```

Arguments

`OFF` or 0 hides the specified trace.

`ON` or 1 shows the specified trace.

Examples

`TRACE1:PSETTLING:SHOW ON` shows Trace 1 in the Phase Settling Time view.

TRACe<x>:LTE:ACLR:COUNt

Sets or queries how many acquisitions run in the single acquisition mode for multitrace functions (Max Hold and Average) in the LTE ACLR measurement. This command is effective when `TRACe<x>: LTE:ACLR:FUNCtion` is set to `MAXHold` or `AVERage` and `INITiate:CONTinuous` is set to OFF.

Conditions

Measurement view: LTE ACLR

Group

Trace commands

Syntax

```
TRACe<x>:LTE:ACLR:COUNt <number>
```

Related Commands

TRACe<x>:LTE:ACLR:FUNCtion (on page 2655)

Arguments

`<number>::<NR1>` specifies the count for multi-trace functions. Range: 1 to 10000.

Returns

`<NR1>`

Examples

`TRACE1:LTE:ACLR:COUNT 32` sets the count to 32 for multitrace functions.

TRACe<x>:LTE:ACLR:COUNt:RESet (No Query Form)

Clears the multifunction (Max Hold or Average) data and counter and restarts the process in the LTE ACLR display. This command is effective when `TRACe<x>:LTE:ACLR:FUNCtion` is set to `MAXHold` or `AVERage`.

Conditions

Measurement view: LTE ACLR

Group

Trace commands

Syntax

```
TRACe<x>:LTE:ACLR:COUNt:RESet
```

Related Commands

TRACe<x>:LTE:ACLR:FUNCtion (on page 2655)

Arguments

None

Examples

`TRACE1:LTE:ACLR:COUNT:RESET` clears the multifunction data and counter, and restarts the process.

TRACe<x>:LTE:ACLR:FUNCtion

Sets or queries the trace function in the LTE ACLR display.

Conditions

Measurement view: LTE ACLR

Group

Trace commands

Syntax

```
TRACe<x>:LTE:ACLR:FUNCtion { NONE | MAXH | AVER }
TRACe<x>:LTE:ACLR:FUNCtion?
```

Arguments

`NONE` sets Normal display of Trace 1.

`MAXHold` sets Max Hold display of Trace 1, which indicates the maximum amplitude drift at each frequency point.

`AVERage` sets Average display of Trace 1, which indicates the average amplitude drift at each frequency point.

Returns

`NONE` means that Trace 1 is set to Normal display. `MAXHold` means Trace 1 is set to Max Hold display. `AVERage` means Trace 1 is set to Average display.

Examples

`TRACE1:LTE:ACLR:FUNCTION MAXH` displays the Max Hold trace in the LTE ACLR display.

TRACe<x>:LTE:CONSte:SYNChronization:SEQUence:STATe? (Query Only)

Returns whether or not a measurement signal (Primary Synchronization Signal/Secondary Synchronization Signal) is selected for the specified trace in the Trace tab of the settings control panel of the LTE Constellation display.

Conditions

Measurement view: LTE Constellation

Group

Trace commands

Syntax

```
TRACe<x>:LTE:CONSte:SYNChronization:SEQUence:STATe?
```

Returns

0 means no measurement signal is selected.

1 means a measurement signal is selected.

Examples

`TRACE1:LTE:CONSTE:SYNCHRONIZATION:SEQUENCE:STATE?` might return `1`, indicating that SSS or PSS is selected in the Trace tab of the LTE Constellation display.

TRACe<x>:NOISe:FIGure:AVERage:COUNt

Sets or queries the average counts for the specified trace in the Noise Figure display.

Conditions

Measurement view: Noise Figure

The parameter <x>= 1 to 3.

Group

Trace commands

Syntax

```
TRACe<x>:NOISe:FIGure:AVERage:COUNt <NR1>
TRACe<x>:NOISe:FIGure:AVERage:COUNt?
```

Arguments

`<NR1> = integer`

Range: 2 to 10000

Returns

`<NR3>` floating point value with an exponent, which is the average count value.

Examples

`TRAC1:NOIS:FIG:AVER:COUN?` might return `10.0000000000E+3`, indicating that the average count value is set to 10000.

TRACe<x>:NOISe:FIGure:COUNt:ENABle

Sets or queries whether or not the count feature for the specified trace in the Noise Figure display is enabled or disabled.

Conditions

Measurement view: Noise Figure display The parameter <x>= 1 to 3.

Only available when trace Function is set to Avg (VRMS), Max Hold, or Min Hold.

Group

Trace commands

Syntax

```
TRACe<x>:NOISe:FIGure:COUNt:ENABle { OFF | ON | 0 | 1 }
TRACe<x>:NOISe:FIGure:COUNt:ENABle?
```

Arguments

`OFF` or 0 disables the count feature for the specified trace.

`ON` or 1 enables the count feature for the specified trace.

Returns

0 indicates the count feature for the specified trace if disabled.

1 indicates the count feature for the specified trace if enabled.

Examples

`TRAC1:NOIS:FIG:COUN:ENAB ON` enables the count feature for Trace 1.

`TRAC1:NOIS:FIG:COUN:ENAB?` might return `0`, indicating that the count feature for Trace 1 is disabled.

TRACe<x>:NOISe:FIGure:DETection

Sets or queries the trace detection method in the Noise Figure display.

Conditions

Measurement view: Noise Figure display The parameter <x> = 1 to 3.

Group

Trace commands

Syntax

```
TRACe<x>:NOISe:FIGure:DETection { AVERage }
TRACe<x>:NOISe:FIGure:DETection?
```

Arguments

`AVERage`: Each point on the trace is the result of determining the RMS Voltage value for all of the IQ samples available to the trace point. When displayed in either linear (Volts, Watts) or Log (dB, dBm), the correct RMS value results.

When the averaging function is applied to a trace, the averaging is performed on the linear (Voltage) values, resulting in the correct average for RMS values.

Returns

`AVER` means that the trace function is set to Average (VRMS).

Examples

`TRACE1:NOIS:FIG:DET AVER` sets the detection method for Trace 1 to Average (VRMS).

`TRACE2:NOIS:FIG:DET?` might return `AVER`, indicating that the function for Trace 1 is set to Average (VRMS).

TRACe<x>:NOISe:FIGure:FREeze

Specifies whether or not to freeze (halt updates of) the specified trace in the Noise Figure display.

Conditions

Measurement view: Noise Figure display

The parameter <x>= 1 to 3.

Group

Trace commands

Syntax

```
TRACe<x>:NOISe:FIGure:FREeze { OFF | ON | 0 | 1 }
TRACe<x>:NOISe:FIGure:FREeze?
```

Arguments

`OFF` or 0 normally updates the display of the specified trace.

`ON` or 1 stops updating the display of the specified trace.

Returns

0 indicates the specified trace is being updated normally in the Noise Figure display.

1 indicates the specified trace is not being updated in the Noise Figure display.

Examples

`TRAC1:NOIS:FIG:FRE ON` freezes the display of Trace 1.

`TRAC1:NOIS:FIG:FRE?` might return `0`, indicating that the display of Trace 1 is set to Freeze.

TRACe<x>:NOISe:FIGure:FUNCtion

Sets or queries the trace processing method (function) in the Noise Figure display.

Conditions

Measurement view: Noise Figure display

The parameter <x> = 1 to 3.

Group

Trace commands

Syntax

```
TRACe<x>:NOISe:FIGure:FUNCtion { NORMal | AVERage | MAXHold| MINHold}
TRACe<x>:NOISe:FIGure:FUNCtion?
```

Arguments

`NORMal`: Displays the trace record for each display point without additional processing.

AVERage: Each point on the trace is the result of determining the RMS Voltage value for all of the IQ samples available to the trace point. When displayed in either linear (Volts, Watts) or Log (dB, dBm), the correct RMS value results.

When the averaging function is applied to a trace, the averaging is performed on the linear (Voltage) values, resulting in the correct average for RMS values.

MAXHold: Displays the maximum value in the trace record for each display point.

MINHold: Displays the minimum value in the trace record for each display point.

Returns

NORM means that the trace function is set to Normal.

AVER means that the trace function is set to Average (VRMS).

MAXH means that the trace function is set to Max Hold.

MINH means that the trace function is set to Min Hold.

Examples

TRAC1:NOIS:FIG:FUNC NORM sets the function for Trace 1 to Normal.

TRAC2:NOIS:FIG:FUNC? might return MAXH, indicating that the function for Trace 1 is set to Max Hold.

TRACe<x>:NOISe:FIGure:SELect

Sets the trace to be shown in the Noise Figure display. The query form of this command returns whether or not the specified trace is set to show (ON) or not (OFF).

Conditions

Measurement view: Noise Figure display The parameter <x> = 1 to 3.

Group

Trace commands

Syntax

```
TRACe<x>:NOISe:FIGure:SELect
TRACe<x>:NOISe:FIGure:SELect?
```

Arguments

None

Returns

0 means that the specified trace is not shown (set to Off).

1 means that the specified trace is set to show (set to normal).

Examples

TRAC1:NOIS:FIG:SEL will set the display to show Trace 1.

TRAC2:NOIS:FIG:SEL? might return 1, indicating that Trace 2 is set to show.

TRACe<x>:NOISe:FIGure:SHOW

Sets or queries the state (shown or not shown) of the specified trace in the Noise Figure display.

Conditions

Measurement view: Noise Figure display The parameter <x> = 1 to 3.

Group

Trace commands

Syntax

```
TRACe<x>:NOISe:FIGure:SHOW { OFF | ON | 0 | 1 }
TRACe<x>:NOISe:FIGure:SHOW?
```

Arguments

`OFF` or 0 hides the specified trace.

`ON` or 1 displays the specified trace.

Returns

0 indicates the specified trace is not shown in the Noise Figure display.

1 indicates the specified trace is shown in the Noise Figure display.

Examples

`TRAC1:NOIS:FIG:SHOW` sets Trace to show in the Noise Figure display.

`TRAC2:NOIS:FIG:SHOW?` might return `0`, indicating that Trace 2 is not showing in the Noise Figure display.

TRACe<x>:NOISe:GAIN:AVERage:COUNt

Sets or queries the average counts for the specified trace in the Gain display.

Conditions

Measurement view: Gain

The parameter <x>= 1 to 3.

Group

Trace commands

Syntax

```
TRACe<x>:NOISe:GAIN:AVERage:COUNt <NR1>
TRACe<x>:NOISe:GAIN:AVERage:COUNt?
```

Arguments

`<NR1> = integer`

Range: 2 to 10000

Returns

`<NR3>` floating point value with an exponent, which is the average count value.

Examples

`TRAC1:NOIS:GAIN:AVER:COUN?` might return `10.0000000000E+3`, indicating that the average count value is set to 10000.

TRACe<x>:NOISe:GAIN:COUNt:ENABle

Sets or queries whether or not the count feature for the specified trace in the Gain display is enabled or disabled.

Conditions

Measurement view: Gain display

The parameter <x>= 1 to 3.

Only available when trace Function is set to Avg (VRMS), Max Hold, or Min Hold.

Group

Trace commands

Syntax

```
TRACe<x>:NOISe:GAIN:COUNt:ENABle { OFF | ON | 0 | 1 }
TRACe<x>:NOISe:GAIN:COUNt:ENABle?
```

Arguments

`OFF` or 0 disables the count feature for the specified trace.

`ON` or 1 enables the count feature for the specified trace.

Returns

0 indicates the count feature for the specified trace if disabled.

1 indicates the count feature for the specified trace if enabled.

Examples

`TRAC1:NOIS:GAIN:COUN:ENAB ON` enables the count feature for Trace 1.

`TRAC1:NOIS:GAIN:COUN:ENAB?` might return `0`, indicating that the count feature for Trace 1 is disabled.

TRACe<x>:NOISe:GAIN:DETection

Sets or queries the trace detection method in the Gain display.

Conditions

Measurement view: Gain display

The parameter <x> = 1 to 3.

Group

Trace commands

Syntax

```
TRACe<x>:NOISe:GAIN:DETection { AVERage }
TRACe<x>:NOISe:GAIN:DETection?
```

Arguments

`AVERage`: Each point on the trace is the result of determining the RMS Voltage value for all of the IQ samples available to the trace point. When displayed in either linear (Volts, Watts) or Log (dB, dBm), the correct RMS value results.

When the averaging function is applied to a trace, the averaging is performed on the linear (Voltage) values, resulting in the correct average for RMS values.

Returns

AVER means that the trace function is set to Average (VRMS).

Examples

`TRACE1:NOIS:YFAC:DET AVER` sets the detection method for Trace 1 to Average (VRMS).

`TRACE2:NOIS:YFAC:DET?` might return `AVER`, indicating that the function for Trace 1 is set to Average (VRMS).

TRACe<x>:NOISe:GAIN:FREeze

Specifies whether or not to freeze (halt updates of) the specified trace in the Gain display.

Conditions

Measurement view: Gain display

The parameter <x>= 1 to 3.

Group

Trace commands

Syntax

```
TRACe<x>:NOISe:GAIN:FREeze { OFF | ON | 0 | 1 }
TRACe<x>:NOISe:GAIN:FREeze?
```

Arguments

`OFF` or 0 normally updates the display of the specified trace.

`ON` or 1 stops updating the display of the specified trace.

Returns

0 indicates the specified trace is being updated normally in the Gain display.

1 indicates the specified trace is not being updated in the Gain display.

Examples

`TRAC1:NOIS:GAIN:FRE ON` freezes the display of Trace 1.

`TRAC1:NOIS:GAIN:FRE?` might return `0`, indicating that the display of Trace 1 is set to Freeze.

TRACe<x>:NOISe:GAIN:FUNCtion

Sets or queries the trace processing method (function) in the Gain display.

Conditions

Measurement view: Gain display

The parameter <x> = 1 to 3.

Group

Trace commands

Syntax

```
TRACe<x>:NOISe:GAIN:FUNCtion { NORMal | AVERage | MAXHold| MINHold}
TRACe<x>:NOISe:GAIN:FUNCtion?
```

Arguments

NORMal: Displays the trace record for each display point without additional processing.

AVERage: Each point on the trace is the result of determining the RMS Voltage value for all of the IQ samples available to the trace point. When displayed in either linear (Volts, Watts) or Log (dB, dBm), the correct RMS value results.

When the averaging function is applied to a trace, the averaging is performed on the linear (Voltage) values, resulting in the correct average for RMS values.

MAXHold: Displays the maximum value in the trace record for each display point.

MINHold: Displays the minimum value in the trace record for each display point.

Returns

NORM means that the trace function is set to Normal.

AVER means that the trace function is set to Average (VRMS).

MAXH means that the trace function is set to Max Hold.

MINH means that the trace function is set to Min Hold.

Examples

TRAC1:NOIS:GAIN:FUNC NORM sets the function for Trace 1 to Normal.

TRAC2:NOIS:GAIN:FUNC? might return MAXH, indicating that the function for Trace 1 is set to Max Hold.

TRACe<x>:NOISe:GAIN:SELect

Sets the trace to be shown in the Gain display. The query form of this command returns whether or not the specified trace is set to show (ON) or not (OFF).

Conditions

Measurement view: Gain display

The parameter <x> = 1 to 3.

Group

Trace commands

Syntax

```
TRACe<x>:NOISe:GAIN:SELect
TRACe<x>:NOISe:GAIN:SELect?
```

Arguments

None

Returns

0 means that the specified trace is not shown (set to Off).

1 means that the specified trace is set to show (set to normal).

Examples

TRAC1:NOIS:GAIN:SEL will set the display to show Trace 1.

TRAC2:NOIS:GAIN:SEL? might return 1, indicating that Trace 2 is set to show.

TRACe<x>:NOISe:GAIN:SHOW

Sets or queries the state (shown or not shown) of the specified trace in the Gain display.

Conditions

Measurement view: Gain display

The parameter <x> = 1 to 3.

Group

Trace commands

Syntax

```
TRACe<x>:NOISe:GAIN:SHOW { OFF | ON | 0 | 1 }
TRACe<x>:NOISe:GAIN:SHOW?
```

Arguments

`OFF` or 0 hides the specified trace.

`ON` or 1 displays the specified trace.

Returns

0 indicates the specified trace is not shown in the Gain display.

1 indicates the specified trace is shown in the Gain display.

Examples

`TRAC1:NOIS:GAIN:SHOW` sets Trace to show in the Gain display.

`TRAC2:NOIS:GAIN:SHOW?` might return `0`, indicating that Trace 2 is not showing in the Gain display.

TRACe<x>:NOISe:TEMPerature:AVERage:COUNt

Sets or queries the average counts for the specified trace in the Noise Temperature display.

Conditions

Measurement view: Noise Temperature The parameter <x>= 1 to 3.

Group

Trace commands

Syntax

```
TRACe<x>:NOISe:TEMPerature:AVERage:COUNt <NR1>
TRACe<x>:NOISe:TEMPerature:AVERage:COUNt?
```

Arguments

`<NR1> = integer`

Range: 2 to 10000

Returns

`<NR3>` floating point value with an exponent, which is the average count value.

Examples

TRAC1:NOIS:TEMP:AVER:COUN? might return 10.0000000000E+3, indicating that the average count value is set to 10000.

TRACe<x>:NOISe:TEMPerature:COUNt:ENABle

Sets or queries whether or not the count feature for the specified trace in the Noise Temperature display is enabled or disabled.

Conditions

Measurement view: Noise Temperature display The parameter <x>= 1 to 3.

Only available when trace Function is set to Avg (VRMS), Max Hold, or Min Hold.

Group

Trace commands

Syntax

```
TRACe<x>:NOISe:TEMPerature:COUNt:ENABle { OFF | ON | 0 | 1 }
TRACe<x>:NOISe:TEMPerature:COUNt:ENABle?
```

Arguments

OFF or 0 disables the count feature for the specified trace.

ON or 1 enables the count feature for the specified trace.

Returns

0 indicates the count feature for the specified trace if disabled.

1 indicates the count feature for the specified trace if enabled.

Examples

TRAC1:NOIS:TEMP:COUN:ENAB ON enables the count feature for Trace 1.

TRAC1:NOIS:TEMP:COUN:ENAB? might return 0, indicating that the count feature for Trace 1 is disabled.

TRACe<x>:NOISe:TEMPerature:DETection

Sets or queries the trace detection method in the Noise Temperature display.

Conditions

Measurement view: Noise Temperature display The parameter <x> = 1 to 3.

Group

Trace commands

Syntax

```
TRACe<x>:NOISe:TEMPerature:DETection { AVERage }
TRACe<x>:NOISe:TEMPerature:DETection?
```

Arguments

AVERage: Each point on the trace is the result of determining the RMS Voltage value for all of the IQ samples available to the trace point. When displayed in either linear (Volts, Watts) or Log (dB, dBm), the correct RMS value results.

When the averaging function is applied to a trace, the averaging is performed on the linear (Voltage) values, resulting in the correct average for RMS values.

Returns

AVER means that the trace function is set to Average (VRMS).

Examples

TRAC1:NOIS:TEMP:DET AVER sets the detection method for Trace 1 to Average (VRMS).

TRAC2:NOIS:TEMP:DET? might return AVER, indicating that the function for Trace 1 is set to Average (VRMS).

TRACe<x>:NOISE:TEMPerature:FREeze

Specifies whether or not to freeze (halt updates of) the specified trace in the Noise Temperature display.

Conditions

Measurement view: Noise Temperature display The parameter <x>= 1 to 3.

Group

Trace commands

Syntax

```
TRACe<x>:NOISE:TEMPerature:FREeze { OFF | ON | 0 | 1 }
TRACe<x>:NOISE:TEMPerature:FREeze?
```

Arguments

OFF or 0 normally updates the display of the specified trace.

ON or 1 stops updating the display of the specified trace.

Returns

0 indicates the specified trace is being updated normally in the Noise Temperature display.

1 indicates the specified trace is not being updated in the Noise Temperature display.

Examples

TRAC1:NOIS:TEMP:FRE ON freezes the display of Trace 1.

TRAC1:NOIS:TEMP:FRE? might return 0, indicating that the display of Trace 1 is set to Freeze.

TRACe<x>:NOISe:TEMPerature:FUNCtion

Sets or queries the trace processing method (function) in the Noise Temperature display.

Conditions

Measurement view: Noise Temperature display The parameter <x> = 1 to 3.

Group

Trace commands

Syntax

```
TRACe<x>:NOISe:TEMPerature:FUNCtion { NORMal | AVERage | MAXHold | MINHold}
TRACe<x>:NOISe:TEMPerature:FUNCtion?
```

Arguments

NORMal: Displays the trace record for each display point without additional processing.

AVERage: Each point on the trace is the result of determining the RMS Voltage value for all of the IQ samples available to the trace point. When displayed in either linear (Volts, Watts) or Log (dB, dBm), the correct RMS value results.

When the averaging function is applied to a trace, the averaging is performed on the linear (Voltage) values, resulting in the correct average for RMS values.

MAXHold: Displays the maximum value in the trace record for each display point.

MINHold: Displays the minimum value in the trace record for each display point.

Returns

NORM means that the trace function is set to Normal.

AVER means that the trace function is set to Average (VRMS).

MAXH means that the trace function is set to Max Hold.

MINH means that the trace function is set to Min Hold.

Examples

TRAC1:NOIS:TEMP:FUNC NORM sets the function for Trace 1 to Normal.

TRAC2:NOIS:TEMP:FUNC? might return MAXH, indicating that the function for Trace 1 is set to Max Hold.

TRACe<x>:NOISe:TEMPerature:SELect

Sets the trace to be shown in the Noise Temperature display. The query form of this command returns whether or not the specified trace is set to show (On) or not (Off).

Conditions

Measurement view: Noise Temperature display The parameter <x> = 1 to 3.

Group

Trace commands

Syntax

```
TRACe<x>:NOISe:TEMPerature:SELect
TRACe<x>:NOISe:TEMPerature:SELect?
```

Arguments

None

Returns

0 means that the specified trace is not shown (set to Off).

1 means that the specified trace is set to show (set to normal).

Examples

TRAC1:NOIS:TEMP:SEL will set the display to show Trace 1.

TRAC2:NOIS:TEMP:SEL? might return 1, indicating that Trace 2 is set to show.

TRACe<x>:NOISe:TEMPerature:SHOW

Sets or queries the state (shown or not shown) of the specified trace in the Noise Temperature display.

Conditions

Measurement view: Noise Temperature display The parameter <x> = 1 to 3.

Group

Trace commands

Syntax

```
TRACe<x>:NOISe:TEMPerature:SHOW { OFF | ON | 0 | 1 }
TRACe<x>:NOISe:TEMPerature:SHOW?
```

Arguments

`OFF` or 0 hides the specified trace.

`ON` or 1 displays the specified trace.

Returns

0 indicates the specified trace is not shown in the Noise Temperature display.

1 indicates the specified trace is shown in the Noise Temperature display.

Examples

`TRAC1:NOIS:TEMP:SHOW` sets Trace 1 to show in the Noise Temperature display.

`TRAC2:NOIS:TEMP:SHOW?` might return `0`, indicating that Trace 2 is not showing in the Noise Temperature display.

TRACe<x>:NOISe:YFACtor:AVERage:COUNt

Sets or queries the average counts for the specified trace in the Y Factor display.

Conditions

Measurement view: Y Factor

The parameter <x>= 1 to 3.

Group

Trace commands

Syntax

```
TRACe<x>:NOISe:YFACtor:AVERage:COUNt <NR1>
TRACe<x>:NOISe:YFACtor:AVERage:COUNt?
```

Arguments

`<NR1> = integer`

Range: 2 to 10000

Returns

`<NR3>` floating point value with an exponent, which is the average count value.

Examples

`TRAC1:NOIS:TEMP:AVER:COUN?` might return `10.0000000000E+3`, indicating that the average count value is set to 10000.

TRACe<x>:NOISe:YFACtor:COUNt:ENABle

Sets or queries whether or not the count feature for the specified trace in the Y Factor display is enabled or disabled.

Conditions

Measurement view: Y Factor display The parameter <x>= 1 to 3.

Only available when trace Function is set to Avg (VRMS), Max Hold, or Min Hold.

Group

Trace commands

Syntax

```
TRACe<x>:NOISe:YFACtor:COUNt:ENABle { OFF | ON | 0 | 1 }
TRACe<x>:NOISe:YFACtor:COUNt:ENABle?
```

Arguments

`OFF` or 0 disables the count feature for the specified trace.

`ON` or 1 enables the count feature for the specified trace.

Returns

0 indicates the count feature for the specified trace if disabled.

1 indicates the count feature for the specified trace if enabled.

Examples

`TRAC1:NOIS:YFAC:COUN:ENAB ON` enables the count feature for Trace 1.

`TRAC1:NOIS:YFAC:COUN:ENAB?` might return `0`, indicating that the count feature for Trace 1 is disabled.

TRACe<x>:NOISe:YFACtor:DETection

Sets or queries the trace detection method in the Y Factor display.

Conditions

Measurement view: Y Factor display The parameter <x> = 1 to 3.

Group

Trace commands

Syntax

```
TRACe<x>:NOISe:YFACtor:DETection { AVERage }
TRACe<x>:NOISe:YFACtor:DETection?
```

Arguments

`AVERage`: Each point on the trace is the result of determining the RMS Voltage value for all of the IQ samples available to the trace point. When displayed in either linear (Volts, Watts) or Log (dB, dBm), the correct RMS value results.

When the averaging function is applied to a trace, the averaging is performed on the linear (Voltage) values, resulting in the correct average for RMS values.

Returns

`AVER` means that the trace function is set to Average (VRMS).

Examples

`TRACE1:NOIS:YFAC:DET AVER` sets the detection method for Trace 1 to Average (VRMS).

`TRACE2:NOIS:YFAC:DET?` might return `AVER`, indicating that the function for Trace 1 is set to Average (VRMS).

TRACe<x>:NOISE:YFACtor:FREeze

Specifies whether or not to freeze (halt updates of) the specified trace in the Y Factor display.

Conditions

Measurement view: Y Factor display The parameter <x>= 1 to 3.

Group

Trace commands

Syntax

```
TRACe<x>:NOISE:YFACtor:FREeze { OFF | ON | 0 | 1 }
TRACe<x>:NOISE:YFACtor:FREeze?
```

Arguments

`OFF` or 0 normally updates the display of the specified trace.

`ON` or 1 stops updating the display of the specified trace.

Returns

0 indicates the specified trace is being updated normally in the Y Factor display.

1 indicates the specified trace is not being updated in the Y Factor display.

Examples

`TRAC1:NOIS:YFAC:FRE ON` freezes the display of Trace 1.

`TRAC1:NOIS:YFAC:FRE?` might return `0`, indicating that the display of Trace 1 is set to Freeze.

TRACe<x>:NOISe:YFACtor:FUNCtion

Sets or queries the trace processing method (function) in the Y Factor display.

Conditions

Measurement view: Y Factor display The parameter <x> = 1 to 3.

Group

Trace commands

Syntax

```
TRACe<x>:NOISe:YFACtor:FUNCtion { NORMal | AVERage | MAXHold| MINHold}
TRACe<x:NOISe:YFACtor:FUNCtion?
```

Arguments

`NORMal`: Displays the trace record for each display point without additional processing.

`AVERage`: Each point on the trace is the result of determining the RMS Voltage value for all of the IQ samples available to the trace point. When displayed in either linear (Volts, Watts) or Log (dB, dBm), the correct RMS value results.

When the averaging function is applied to a trace, the averaging is performed on the linear (Voltage) values, resulting in the correct average for RMS values.

`MAXHold`: Displays the maximum value in the trace record for each display point.

`MINHold`: Displays the minimum value in the trace record for each display point.

Returns

`NORM` means that the trace function is set to Normal.

`AVER` means that the trace function is set to Average (VRMS).

`MAXH` means that the trace function is set to Max Hold.

`MINH` means that the trace function is set to Min Hold.

Examples

`TRAC1:NOIS:YFAC:FUNC NORM` sets the function for Trace 1 to Normal.

`TRAC2:NOIS:YFAC:FUNC?` might return `MAXH`, indicating that the function for Trace 1 is set to Max Hold.

TRACe<x>:NOISe:YFACtor:SELect

Sets the trace to be shown in the Y Factor display. The query form of this command returns whether or not the specified trace is set to show (ON) or not (OFF).

Conditions

Measurement view: Y Factor display The parameter <x> = 1 to 3.

Group

Trace commands

Syntax

```
TRACe<x>:NOISe:YFACtor:SELect
TRACe<x>:NOISe:YFACtor:SELect?
```

Arguments

None

Returns

`0` means that the specified trace is not shown (set to Off).

`1` means that the specified trace is set to show (set to normal).

Examples

`TRAC1:NOIS:YFAC:SEL` will set the display to show Trace 1.

`TRAC2:NOIS:YFAC:SEL?` might return `1`, indicating that Trace 2 is set to show.

TRACe<x>:NOISe:YFACtor:SHOW

Sets or queries the state (shown or not shown) of the specified trace in the Y Factor display.

Conditions

Measurement view: Y Factor display The parameter <x> = 1 to 3.

Group

Trace commands

Syntax

```
TRACe<x>:NOISe:YFACtor:SHOW { OFF | ON | 0 | 1 }
TRACe<x>:NOISe:YFACtor:SHOW?
```

Arguments

`OFF` or 0 hides the specified trace.

`ON` or 1 displays the specified trace.

Returns

0 indicates the specified trace is not shown in the Y Factor display.

1 indicates the specified trace is shown in the Y Factor display.

Examples

`TRACE1:NOISE:YFACTOR:SHOW` sets Trace 1 to show in the Y Factor display.

`TRACE2:NOISE:YFACTOR:SHOW?` might return `0`, indicating that Trace 2 is not showing in the Y Factor display.

TRACe<x>:OFDM:EVM

Determines whether or not to show the specified trace in the OFDM EVM view.

The parameter <x> = 1 or 2, where 1 is Average and 2 is Value.

Conditions

Measurement views: OFDM

Group

Trace commands

Syntax

```
TRACe<x>:OFDM:EVM { OFF | ON | 0 | 1 }
TRACe<x>:OFDM:EVM?
```

Arguments

`OFF` or 0 hides the specified trace in the view.

`ON` or 1 shows the specified trace in the view.

Examples

`TRACe1:OFDM:EVM ON` shows Trace 1 in the OFDM EVM view.

TRACe<x>:OFDM:EVM:FREeze

Determines whether or not to freeze the display of the specified trace in the OFDM EVM view.

The parameter <x> = 1 or 2, where 1 is Average and 2 is Value.

Conditions

Measurement views: OFDM

Group

Trace commands

Syntax

```
TRACe<x>:OFDM:EVM:FREeze { OFF | ON | 0 | 1 }
TRACe<x>:OFDM:EVM:FREeze?
```

Arguments

OFF or 0 updates the display of the specified trace normally.

ON or 1 stops updating the display of the specified trace.

Examples

TRACE1:OFDM:EVM:FREEZE ON freezes the display for Trace 1.

TRACe<x>:OFDM:EVM:SELect

Sets or queries the specified trace to display in the OFDM EVM view. The parameter <x> = 1 or 2, where 1 is Average and 2 is Value.

Conditions

Measurement views: OFDM

Group

Trace commands

Syntax

```
TRACe<x>:OFDM:EVM:SELect
TRACe<x>:OFDM:EVM:SELect?
```

Arguments

None

Returns

{ OFF | ON | 0 | 1 }

OFF or 0 indicates that the readout is not being displayed for the specified trace.

ON or 1 indicates that the readout is being displayed for the specified trace.

Examples

TRACE1:OFDM:EVM:SELECT selects Trace 1 (Average trace) to display the readout.

TRACE1:OFDM:EVM:SELECT? might return 1, indicating that the readout is being displayed for Trace 1.

TRACe<x>:OFDM:MERRor

Determines whether or not to show the specified trace in the OFDM Magnitude error view.

The parameter <x> = 1 or 2, where 1 is Average and 2 is Value.

Conditions

Measurement views: OFDM

Group

Trace commands

Syntax

```
TRACe<x>:OFDM:MERRor { OFF | ON | 0 | 1 }
TRACe<x>:OFDM:MERRor?
```

Arguments

`OFF` or 0 hides the specified trace in the view.

`ON` or 1 shows the specified trace in the view.

Examples

`TRACe1:OFDM:MERRor ON` shows Trace 1 in the OFDM Magnitude error view.

TRACe<x>:OFDM:MERRor:FREeze

Determines whether or not to freeze the display of the specified trace in the OFDM Magnitude error view.

The parameter <x> = 1 or 2, where 1 is Average and 2 is Value.

Conditions

Measurement views: OFDM

Group

Trace commands

Syntax

```
TRACe<x>:OFDM:MERRor:FREeze { OFF | ON | 0 | 1 }
TRACe<x>:OFDM:MERRor:FREeze?
```

Arguments

`OFF` or 0 updates the display of the specified trace normally.

`ON` or 1 stops updating the display of the specified trace.

Examples

`TRACE1:OFDM:MERRor:FREEZE ON` freezes the display for Trace 1.

TRACe<x>:OFDM:MERRor:SELect

Sets or queries the specified trace to display in the OFDM Magnitude error view. The parameter <x> = 1 or 2, where 1 is Average and 2 is Value.

Conditions

Measurement views: OFDM

Group

Trace commands

Syntax

```
TRACe<x>:OFDM:MERRor:SELect { OFF | ON | 0 | 1 }
TRACe<x>:OFDM:MERRor:SELect?
```

Arguments

None

Returns

`{ OFF | ON | 0 | 1 }`

`OFF` or 0 indicates that the readout is not being displayed for the specified trace.

`ON` or 1 indicates that the readout is being displayed for the specified trace.

Examples

`TRACE1:OFDM:MERRor:SELECT` selects Trace 1 (Average trace) to display the readout.

`TRACE1:OFDM:MERRor:SELECT?` might return `1`, indicating that the readout is being displayed for Trace 1.

TRACe<x>:OFDM:PERRor

Determines whether or not to show the specified trace in the OFDM Phase error view.

The parameter <x> = 1 or 2, where 1 is Average and 2 is Value.

Conditions

Measurement views: OFDM

Group

Trace commands

Syntax

```
TRACe<x>:OFDM:PERRor { OFF | ON | 0 | 1 }
TRACe<x>:OFDM:PERRor?
```

Arguments

`OFF` or 0 hides the specified trace in the view.

`ON` or 1 shows the specified trace in the view.

Examples

`TRACe1:OFDM:PERRor ON` shows Trace 1 in the OFDM Phase error view.

TRACe<x>:OFDM:PERRor:FREeze

Determines whether or not to freeze the display of the specified trace in the OFDM Phase error view.

The parameter <x> = 1 or 2, where 1 is Average and 2 is Value.

Conditions

Measurement views: OFDM

Group

Trace commands

Syntax

```
TRACe<x>:OFDM:PERRor:FREeze { OFF | ON | 0 | 1 }
TRACe<x>:OFDM:PERRor:FREeze?
```

Arguments

OFF or 0 updates the display of the specified trace normally.

ON or 1 stops updating the display of the specified trace.

Examples

TRACE1:OFDM:PERRor:FREEZE ON freezes the display for Trace 1.

TRACe<x>:OFDM:PERRor:SELect

Sets or queries the specified trace to display in the OFDM Phase error view. The parameter <x> = 1 or 2, where 1 is Average and 2 is Value.

Conditions

Measurement views: OFDM

Group

Trace commands

Syntax

```
TRACe<x>:OFDM:PERRor:SELect { OFF | ON | 0 | 1 }
TRACe<x>:OFDM:PERRor:SELect?
```

Arguments

None

Returns

{ OFF | ON | 0 | 1 }

OFF or 0 indicates that the readout is not being displayed for the specified trace.

ON or 1 indicates that the readout is being displayed for the specified trace.

Examples

TRACE1:OFDM:PERRor:SELECT selects Trace 1 (Average trace) to display the readout.

TRACE1:OFDM:PERRor:SELECT? might return 1, indicating that the readout is being displayed for Trace 1.

TRACe<x>:OFDM:POWer

Determines whether or not to show the specified trace in the OFDM Power view. The parameter <x> = 1 or 2, where 1 is Average and 2 is Value.

Conditions

Measurement views: OFDM

Group

Trace commands

Syntax

```
TRACe<x>:OFDM:POWer { OFF | ON | 0 | 1 }
TRACe<x>:OFDM:POWer?
```

Arguments

OFF or 0 hides the specified trace in the view.

ON or 1 shows the specified trace in the view.

Examples

TRACe1:OFDM:POWer ON shows Trace 1 in the OFDM Power view.

TRACe<x>:OFDM:POWer:FREeze

Determines whether or not to freeze the display of the specified trace in the OFDM Power view.

The parameter <x> = 1 or 2, where 1 is Average and 2 is Value.

Conditions

Measurement views: OFDM

Group

Trace commands

Syntax

```
TRACe<x>:OFDM:POWer:FREeze { OFF | ON | 0 | 1 }
TRACe<x>:OFDM:POWer:FREeze?
```

Arguments

OFF or 0 updates the display of the specified trace normally.

ON or 1 stops updating the display of the specified trace.

Examples

TRACE1:OFDM:POWer:FREEZE ON freezes the display for Trace 1.

TRACe<x>:OFDM:POWer:SELect

Sets or queries the specified trace to display in the OFDM Power view. The parameter <x> = 1 or 2, where 1 is Average and 2 is Value.

Conditions

Measurement views: OFDM

Group

Trace commands

Syntax

```
TRACe<x>:OFDM:POWer:SELect { OFF | ON | 0 | 1 }
TRACe<x>:OFDM:POWer:SELect?
```

Arguments

None

Returns

{ OFF | ON | 0 | 1 }

OFF or 0 indicates that the readout is not being displayed for the specified trace.

ON or 1 indicates that the readout is being displayed for the specified trace.

Examples

TRACE1:OFDM:POWer:SELECT selects Trace 1 (Average trace) to display the readout.

TRACE1:OFDM:POWer:SELECT? might return 1, indicating that the readout is being displayed for Trace 1.

TRACe<x>:P25:CONStellation

Specifies whether or not to show the specified trace (Trace 1 or 2) in the P25 Constellation display.

Conditions

Measurement view: P25 Constellation display The parameter <x>= 1 or 2.

Group

Trace commands

Syntax

```
TRACe<x>:P25:CONStellation { OFF | ON | 0 | 1 }
TRACe<x>:P25:CONStellation?
```

Arguments

OFF or 0 hides the specified trace.

ON or 1 displays the specified trace.

Returns

0 indicates the specified trace is not shown in the P25 Constellation display.

1 indicates the specified trace is shown in the P25 Constellation display.

Examples

TRAC1:P25:CONS ON specifies that Trace 1 be shown in the P25 Constellation display.

TRACe<x>:P25:CONStellation:FREeze

Specifies whether or not to freeze (halt updates of) the specified trace (Trace 1 or 2) in the P25 Constellation display.

Conditions

Measurement view: P25 Constellation display The parameter <x>= 1 or 2.

Group

Trace commands

Syntax

```
TRACe<x>:P25:CONStellation:FREeze { OFF | ON | 0 | 1 }
TRACe<x>:P25:CONStellation:FREeze?
```

Arguments

OFF or 0 normally updates the display of the specified trace.

ON or 1 stops updating the display of the specified trace.

Returns

0 indicates the specified trace is being updated normally in the P25 Constellation display.

1 indicates the specified trace is not being updated in the P25 Constellation display.

Examples

`TRAC1:P25:CONS:FRE ON` freezes the display of Trace 1.

`TRAC1:P25:CONS:FRE?` might return `0`, indicating that the display of Trace 1 is set to Freeze.

TRACe<x>:P25:CONStellation:MODE

Sets or queries how to display the trace content in the P25 Constellation display.

Conditions

Measurement view: P25 Constellation display The parameter <x>= 1 or 2.

Group

Trace commands

Syntax

```
TRACe<x>:P25:CONStellation:MODE { VECT | SYMB | LIN }
```

Arguments

`VECT` (vectors) connects adjacent symbol points with the signal locus.

`SYMB` (symbols) displays individual symbol points.

`LIN` (line) displays all symbol points as a line.

Returns

`VECT` adjacent symbol points are connected with the signal locus.

`SYMB` individual symbol points are displayed.

`LIN` all symbol points are displayed as a line.

Examples

`TRAC1:P25:CONS:MODE SYMB` sets the display to show the constellation of Trace 1 connecting adjacent symbol points with the signal locus.

TRACe<x>:P25:CONStellation:SELect

Selects the specified trace (Trace 1 or 2) in the P25 Constellation display.

Conditions

Measurement view: P25 Constellation display The parameter <x>= 1 or 2.

Group

Trace commands

Syntax

```
TRACe<x>:P25:CONStellation:SELect { OFF | ON | 0 | 1 }
TRACe<x>:P25:CONStellation:SELect?
```

Arguments

None

Returns

0 indicates the specified trace is not the active trace in the P25 Constellation display.

1 indicates the specified trace is the active trace in the P25 Constellation display.

Examples

`TRAC1:P25:CONS:SEL?` might return `1`, indicating that Trace 1 is the selected trace in the P25 Constellation display.

TRACe<x>:P25:EDIagram:ENABle:I

Determines whether to show or hide the I trace in the P25 Eye Diagram display.

Conditions

Measurement view: P25 Eye Diagram

Group

Trace commands

Syntax

```
TRACe<x>:P25:EDIagram:ENABle:I
TRACe<x>:P25:EDIagram:ENABle:I?
```

Related Commands

TRACe<x>:P25:EDIagram:ENABle:Q (on page 2680)

Arguments

`OFF` or 0 hides the I trace.

`ON` or 1 shows the I trace.

Returns

0 indicates the trace is off.

1 indicates the trace is on.

Examples

`TRACEX:P25:EDIAGRAM:ENABLE:I ON` shows the I trace in the P25 Eye Diagram display.

TRACe<x>:P25:EDIagram:ENABle:Q

Determines whether to show or hide the Q trace in the P25 Eye Diagram display.

Conditions

Measurement view: P25 Eye Diagram

Group

Trace commands

Syntax

```
TRACe<x>:P25:EDIagram:ENABle:Q
TRACe<x>:P25:EDIagram:ENABle:Q?
```

Related Commands

TRACe<x>:P25:EDIagram:ENABle:I (on page 2680)

Arguments

`OFF` or 0 hides the Q trace.

`ON` or 1 shows the Q trace.

Returns

0 indicates the trace is off.

1 indicates the trace is on.

Examples

`TRACEX:P25:EDIAGRAM:ENABLE:Q ON` shows the Q trace in the P25 Eye Diagram display.

TRACe<x>:P25:EDIagram:SELect:I

This command selects the I trace in the P25 Eye Diagram display. The query version of this command returns whether the I trace is selected or not.

Conditions

Measurement view: P25 Eye Diagram

Group

Trace commands

Syntax

```
TRACe<x>:P25:EDIagram:SELect:I
TRACe<x>:P25:EDIagram:SELect:I?
```

Related Commands

TRACe<x>:P25:EDIagram:SELect:Q (on page 2682)

Arguments

None.

Returns

`0` indicates that the I trace is not the selected trace.

`1` indicates that the I trace is the selected trace.

Examples

`TRACEX:P25:EDIAGRAM:SELECT:I` selects the I trace in the P25 Eye Diagram display.

TRACe<x>:P25:EDIagram:SELect:Q

This command selects the Q trace in the P25 Eye Diagram display. The query version of this command returns whether the Q trace is selected or not.

Conditions

Measurement view: P25 Eye Diagram

Group

Trace commands

Syntax

```
TRACe<x>:P25:EDIagram:SELect:Q
TRACe<x>:P25:EDIagram:SELect:Q?
```

Related Commands

TRACe<x>:P25:EDIagram:SELect:I (on page 2681)

Arguments

None.

Returns

0 indicates that the Q trace is not the selected trace.

1 indicates that the Q trace is the selected trace.

Examples

`TRACEX:P25:EDIAGRAM:SELECT:Q` selects the Q trace in the P25 Eye Diagram display.

TRACe<x>:PNOise:SELect

Selects the trace in the phase noise measurement. The query returns the currently selected trace.

The parameter <x> = 1 and 2.

Conditions

Measurement views: Phase noise

Group

Trace commands

Syntax

```
TRACe<x>:PNOise:SELect
TRACe<x>:PNOise:SELect?
```

Arguments

None

Returns

0 (not selected) or 1 (selected).

Examples

TRACE2:PNOISE:SELECT selects Trace 2.

TRACe<x>:PNOise:SHOW

Shows or hides the specified trace in the phase noise view. The parameter <x> = 1 and 2.

Conditions

Measurement views: Phase noise

Group

Trace commands

Syntax

```
TRACe<x>:PNOise:SHOW { OFF | ON | 0 | 1 }
TRACe<x>:PNOise:SHOW?
```

Arguments

OFF or 0 hides the specified trace.

ON or 1 shows the specified trace.

Examples

TRACE1:PNOISE:SHOW ON shows Trace 1 in the phase noise view.

TRACe<x>:PNOise:SMOothing:COUNt

Sets or queries the number of data points to take the moving average for smoothing the trace. This command is effective when TRACe<x>:PNOise:SMOothing:ENABle is set to ON.

The parameter <x> = 1 and 2.

Conditions

Measurement views: Phase noise

Group

Trace commands

Syntax

```
TRACe<x>:PNOise:SMOothing:COUNt <number>
TRACe<x>:PNOise:SMOothing:COUNt?
```

Arguments

<number>::<NR1> specifies the number of data points to take the moving average for smoothing. Range: 3 to 50.

Examples

TRACE1:PNOISE:SMOOTHING:COUNT 16 sets the smoothing count to 16 for Trace 1.

TRACe<x>:PNOise:SMOothing:ENABle

Determines whether to enable or disable smoothing the specified trace in the phase noise view.

The parameter <x> = 1 and 2.

Conditions

Measurement views: Phase noise

Group

Trace commands

Syntax

```
TRACe<x>:PNOise:SMOothing:ENABle { OFF | ON | 0 | 1 }
TRACe<x>:PNOise:SMOothing:ENABle?
```

Arguments

`OFF` or 0 disables smoothing.

`ON` or 1 enables smoothing.

Examples

`TRACE1:PNOISE:SMOOTHING:ENABLE ON` enables smoothing Trace 1 in the phase noise view.

TRACe<x>:PNOise:SMOothing:RESet (No Query Form)

Restarts the smoothing process, discarding accumulated data and resetting the counter. This command is effective when `TRACe<x>:PNOise:SMOothing: ENABle` is set to ON.

The parameter <x> = 1 and 2.

Conditions

Measurement views: Phase noise

Group

Trace commands

Syntax

TRACe<x>:PNOise:SMOothing:RESet

Arguments

None

Examples

`TRACE1:PNOISE:SMOOTHING:RESET` restarts the smoothing process for Trace 1.

TRACe<x>:SPECtrum[:MEASview<y>]

Sets or queries the Reference level of measurement source for Q data type.

Conditions

Measurement views: Spectrum

Group

Trace commands

Syntax

```
TRACe<x>:SPECtrum[:MEASview<y>] { OFF | ON | 0 | 1 }
TRACe<x>:SPECtrum[:MEASview<y>]?
```

Arguments

OFF or 0 hides the specified trace.

ON or 1 shows the specified trace.

For Trace 5 (spectrogram), use the TRACe:SGRam:SELect:LINE command to select the line to send to the spectrum display.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter <y>= 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

TRACE1:SPECTRUM:MEASview2 ON shows Trace 1 in the Spectrum Analyzer view.

TRACe<x>:SPECtrum[:MEASview<y>]:AVERage:COUNt

Sets or queries the number of traces to combine. This command is effective when TRACe<x>:SPECtrum[:MEASview<y>]:FUNCtion is AVERage, MAXHold or MINHold.

The parameter <x> = 1 to 3; Trace 4 (math trace) and Trace 5 (spectrogram) are invalid.

Conditions

Measurement views: Spectrum

Group

Trace commands

Syntax

```
TRACe<x>:SPECtrum[:MEASview<y>]:AVERage:COUNt <number>
TRACe<x>:SPECtrum[:MEASview<y>]:AVERage:COUNt?
```

Arguments

<number>::<NR1> specifies the number of traces to combine for averaging.

Range: 1 to 10000.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter <y>= 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

TRACE1:SPECTRUM:MEASVIEW2:AVERAGE:COUNT 64 sets the average count to 64 for Trace 1.

TRACe<x>:SPECtrum[:MEASview<y>]:AVERage:RESet (No Query Form)

Clears average data and counter, and restarts the average process for the specified trace in the Spectrum view. This command is effective when TRACe<x>:SPECtrum[:MEASview<y>]:FUNCtion is set to AVERage, MAXHold or MINHold.

The parameter <x> = 1 to 3; Trace 4 (math trace) and Trace 5 (spectrogram) are invalid.

Conditions

Measurement views: Spectrum

Group

Trace commands

Syntax

`TRACe<x>:SPECtrum[:MEASview<y>]:AVERage:RESet`

Arguments

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

The parameter <y>= 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`TRACE1:SPECTRUM:MEASVIEW2:AVERAGE:RESET` clears average data and counter, and restarts the average process for Trace 1.

TRACe<x>:SPECtrum[:MEASview<y>]:COLor:CURVe

Sets or queries how colors are mapped to the signal density in the Spectrum bitmap display. The mapping can be linear (Curve = 1), or it can be set to concentrate the resolution on the lower level of the range (Curve > 1) or the mapping can be set to show the best resolution on the upper range of density or hit count (Curve = 0.1 - 0.99). This command is only valid for the Bitmap display.

The parameter <x> = 6; only Trace 6 is valid.

Conditions

Measurement views: Spectrum

Group

Trace commands

Syntax

`TRACe<x>:SPECtrum[:MEASview<y>]:COLor:CURVe <value>`

`TRACe<x>:SPECtrum[:MEASview<y>]:COLor:CURVe?`

Arguments

`<value>` = <NRf> specifies how colors are mapped to the signal density. Range: 0.1 to 10 (the default value is 2).

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`TRACe6:SPECTRUM:COLor:CURVe 1.5` concentrates the resolution on the lower level of the range on the Bitmap display.

TRACe<x>:SPECtrum[:MEASview<y>]:COLor:SCALe:AUTO (No Query Form)

Automatically adjusts the Max and Min color settings to display the broadest range of colors in the Spectrum bitmap display. This command is only valid for the Bitmap display.

The parameter <x> = 6; only Trace 6 is valid.

Conditions

Measurement views: Spectrum

Group

Trace commands

Syntax

`TRACe<x>:SPECtrum[:MEASview<y>]:COLor:SCALe:AUTO`

Arguments

`<x>` = 6 specifies the auto-color setting for the Bitmap trace.

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`TRACe6:SPECTRUM:COLor:SCALe:AUTO` auto sets the Bitmap display color scheme.

TRACe<x>:SPECtrum[:MEASview<y>]:COUNt

Sets or queries how many acquisitions run in the single acquisition mode for the Max or Min Hold trace in the Spectrum measurement. This command is effective when `TRACe<x>:SPECtrum[:MEASview<y>]:FUNCtion` is set to `MAXHold` or `MINHold` and `INITiate:CONTinuous` is set to OFF.

The parameter <x> = 1 to 3; Trace 4 (math trace) and Trace 5 (spectrogram) are invalid.

Conditions

Measurement views: Spectrum

Group

Trace commands

Syntax

```
TRACe<x>:SPECtrum[:MEASview<y>]:COUNt <number>
TRACe<x>:SPECtrum[:MEASview<y>]:COUNt?
```

Arguments

`<number>::<NR1>` specifies the count for Max/Min Hold.

Range: 1 to 10000.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

The parameter <y>= 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

TRACE1:SPECTRUM:MEASVIEW2:COUNT 32 sets the count to 32 for Trace 1.

TRACe<x>:SPECtrum[:MEASview<y>]:COUNt:ENABle

Determines whether to enable or disable the count for the Max or Min Hold trace in the Spectrum view. This command is effective when TRACe<x>:SPECtrum[:MEASview<y>]:FUNCtion is set to MAXHold or MINHold.

The parameter <x> = 1 to 3; Trace 4 (math trace) and Trace 5 (spectrogram) are invalid.

Conditions

Measurement views: Spectrum

Group

Trace commands

Syntax

```
TRACe<x>:SPECtrum[:MEASview<y>]:COUNt:ENABle { OFF | ON | 0 | 1 }
TRACe<x>:SPECtrum[:MEASview<y>]:COUNt:ENABle?
```

Arguments

OFF or 0 disables the count for the Max/Min Hold trace.

ON or 1 enables the count for the Max/Min Hold trace.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter <y>= 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

TRACE1:SPECTRUM:MEASVIEW2:COUNT:ENABLE ON enables the Max/Min Hold count for Trace 1.

TRACe<x>:SPECtrum[:MEASview<y>]:COUNt:RESet (No Query Form)

Clears the Max or Min Hold data and counter, and restarts the process for the specified trace in the Spectrum view. This command is effective when TRACe<x>:SPECtrum[:MEASview<y>]:FUNCtion is set to MAXHold or MINHold.

The parameter <x> = 1 to 3; Trace 4 (math trace) and Trace 5 (spectrogram) are invalid.

Conditions

Measurement views: Spectrum

Group

Trace commands

Syntax

```
TRACe<x>:SPECtrum[:MEASview<y>]:COUNt:RESet
```

Arguments

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter <y>= 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

TRACE1:SPECTRUM:MEASVIEW2:COUNT:RESET clears the Max/Min Hold data and counter, and restarts the process for Trace 1.

TRACe<x>:SPECtrum[:MEASview<y>]:DETection

Sets or queries the display detector (method to be used for decimating traces to fit the available horizontal space on screen). The number of horizontal pixels on screen is generally smaller than that of waveform data points. When actually displayed, the waveform data is therefore thinned out, according to the number of pixels, for being compressed.

The parameter <x> = 1 to 3; Trace 4 (math trace) and Trace 5 (spectrogram) are invalid.

Conditions

Measurement views: Spectrum

Group

Trace commands

Syntax

```
TRACe<x>:SPECtrum[:MEASview<y>]:DETection { AVERage | POSitive | NEGative| CAVerage | CPEak |
   QUASipeak | SAMPle | AVGLog }
TRACe<x>:SPECtrum[:MEASview<y>]:DETection?
```

Arguments

AVERage displays the average data value for each pixel.

POSitive displays the maximum data value for each pixel.

NEGative displays the minimum data value for each pixel.

CAVerage displays the CISPR average value for each pixel.

CPEak displays the CISPR peak value for each pixel.

QUASipeak displays the quasi-peak value for each pixel.

SAMPle displays the most recent sample value for each pixel.

AVGLog displays the average data value of logs for each pixel.

The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command

The parameter <y>= 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

TRACE1:SPECTRUM:MEASVIEW2:DETECTION POSitive displays the maximum data value for each pixel on Trace 1

TRACe<x>:SPECtrum[:MEASview<y>]:DOT:PERSistent

Determines whether to enable or disable the dot persistence for the bitmap trace (Trace 6) in the Spectrum view.

The parameter <x> = 6; Only Trace 6 (bitmap trace) is valid.

Conditions

Measurement views: Spectrum

Group

Trace commands

Syntax

`TRACe<x>:SPECtrum[:MEASview<y>]:DOT:PERSistent { OFF | ON | 0 | 1 }`

`TRACe<x>:SPECtrum[:MEASview<y>]:DOT:PERSistent?`

Arguments

`OFF` or `0` disables the dot persistence.

`ON` or `1` enables the dot persistence.

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`TRACE6:SPECTRUM:DOT:PERSISTENT ON` enables the dot persistence in the Spectrum view.

TRACe<x>:SPECtrum[:MEASview<y>]:DOT:PERSistent:TYPE

Sets or queries the persistence type for the bitmap trace (Trace 6) in the Spectrum view.

The parameter <x> = 6; Only Trace 6 (bitmap trace) is valid.

Conditions

Measurement views: Spectrum

Group

Trace commands

Syntax

`TRACe<x>:SPECtrum[:MEASview<y>]:DOT:PERSistent:TYPE { VARiable | INFinite }`

`TRACe<x>:SPECtrum[:MEASview<y>]:DOT:PERSistent:TYPE?`

Arguments

`VARiable`: Selects the variable persistence display which leaves acquired data points on the display for a period of time specified by the `TRACe<x>:SPECtrum[:MEASview<y>]:DOT:PERSistent:VARiable` command.

`INFinite`: Selects the infinite persistence display which accumulates data points on the display indefinitely.

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`TRACE6:SPECTRUM:DOT:PERSISTENT:TYPE VARiable` selects the variable persistence display.

TRACe<x>:SPECtrum[:MEASview<y>]:DOT:PERSistent:VARiable

Sets or queries how long data points are displayed. This command is effective when `TRACe<x>:SPECtrum[:MEASview<y>]:DOT:PERSistent:TYPE` is set to `VARiable`. This affects the display only.

The parameter <x> = 6; Only Trace 6 (bitmap trace) is valid.

Conditions

Measurement views: Spectrum

Group

Trace commands

Syntax

`TRACe<x>:SPECtrum[:MEASview<y>]:DOT:PERSistent:VARiable <number>`

`TRACe<x>:SPECtrum[:MEASview<y>]:DOT:PERSistent:VARiable?`

Arguments

`<number>` = <NRf> specifies the how long waveform points are displayed on the screen. Range: 0 s to 1000 s (the default value is 10 ms).

The parameter <y> = 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`TRACE6:SPECTRUM:DOT:PERSISTENT:VARIABLE 20` specifies that the waveform points are displayed on the screen for a period of 20 before they disappear.

TRACe<x>:SPECtrum[:MEASview<y>]:FREeze

Determines whether or not to freeze the display of the specified trace in the Spectrum view.

The parameter <x> = 1 to 5; All traces are valid.

Conditions

Measurement views: Spectrum

Group

Trace commands

Syntax

```
TRACe<x>:SPECtrum[:MEASview<y>]:FREeze { OFF | ON | 0 | 1 }
TRACe<x>:SPECtrum[:MEASview<y>]:FREeze?
```

Arguments

`OFF` or 0 updates the display of the specified trace normally.

`ON` or 1 stops updating the display of the specified trace.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

The parameter <y>= 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`TRACE1:SPECTRUM:MEASVIEW2:FREEZE ON` freezes Trace 1 in the Spectrum Analysis display.

TRACe<x>:SPECtrum[:MEASview<y>]:FUNCtion

Sets or queries the function for the specified trace in the Spectrum view.

The parameter <x> = 1 to 3; Trace 4 (math trace) and Trace 5 (spectrogram) are invalid.

Conditions

Measurement views: Spectrum

Group

Trace commands

Syntax

```
TRACe<x>:SPECtrum[:MEASview<y>]:FUNCtion { NONE | AVERage | MAXHold | MINHold | AVGLog}
TRACe<x>:SPECtrum[:MEASview<y>]:FUNCtion?
```

Arguments

`NONE` selects the normal spectrum display.

`AVERage` selects the Average display that indicates the average signal level at each frequency point.

`MAXHold` selects the Max Hold display that indicates the maximum signal level at each frequency point.

`MINHold` selects the Min Hold display that indicates the minimum signal level at each frequency point.

`AVGLog` selects the Average of logs display that indicates the average signal level of logs at each frequency point.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

The parameter <y>= 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`TRACE1:SPECTRUM:MEASVIEW2:FUNCTION MAXHold` selects Max Hold for Trace 1 in the Spectrum view.

TRACe<x>:SPECtrum[:MEASview<y>]:LEFToperand

Sets or queries the left operand for the math trace (Trace 4) in the Spectrum view. The parameter <x> = 4; Only Trace 4 (math trace) is valid.

Conditions

Measurement views: Spectrum

Group

Trace commands

Syntax

```
TRACe<x>:SPECtrum[:MEASview<y>]:LEFToperand { TRACE1 | TRACE2 | TRACE3 }
TRACe<x>:SPECtrum[:MEASview<y>]:LEFToperand?
```

Related Commands

TRACe<x>:SPECtrum[:MEASview<y>]:RIGHtoperand (on page 2693)

Arguments

`TRACE1` selects Trace 1 as the left operand for the math trace.

`TRACE2` selects Trace 2 as the left operand for the math trace.

`TRACE3` selects Trace 3 as the left operand for the math trace.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

The parameter <y>= 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`TRACE4:SPECTRUM:MEASVIEW2:LEFTOPERAND TRACE1` selects Trace 1 as the left operand for the math trace.

TRACe<x>:SPECtrum[:MEASview<y>]:RIGHtoperand

Sets or queries the right operand for the math trace (Trace 4) in the Spectrum view. The parameter <x> = 4; Only Trace 4 (math trace) is valid.

Conditions

Measurement views: Spectrum

Group

Trace commands

Syntax

```
TRACe<x>:SPECtrum[:MEASview<y>]:RIGHtoperand { TRACE1 | TRACE2 | TRACE3 }
TRACe<x>:SPECtrum[:MEASview<y>]:RIGHtoperand?
```

Related Commands

TRACe<x>:SPECtrum[:MEASview<y>]:LEFToperand (on page 2692)

Arguments

`TRACE1` selects Trace 1 as the right operand for the math trace.

`TRACE2` selects Trace 2 as the right operand for the math trace.

`TRACE3` selects Trace 3 as the right operand for the math trace.

The parameter <x> = 0 to 4. The specified marker must be activated using the `CALCulate:MARKer:ADD` command

The parameter <y>= 1 to 8 represents measurement index based on number of sources of connected instrument, where MEASview keyword is optional. By default y is 1.

Examples

`TRACE4:SPECTRUM:MEASVIEW2:RIGHTOPERAND TRACE1` selects Trace 1 as the right operand for the math trace

TRACe<x>:SPECtrum[:MEASview<y>]:SELect

Sets or queries the trace to display the readout at the upper left of the Spectrum view.

The parameter <x> = 1 to 5; All traces are valid.

NOTE: TRACe5 (spectrogram) is valid when the spectrum and spectrogram measurements are running.

Conditions

Measurement views: Spectrum

Group

Trace commands

Syntax

```
TRACe<x>:SPECtrum[:MEASview<y>]:SELect
TRACe<x>:SPECtrum[:MEASview<y>]:SELect?
```

Arguments

None

Returns

{ 0 | 1 }

- 0 indicates that the readout is not being displayed for the specified trace.
- 1 indicates that the readout is being displayed for the specified trace.

Examples

`TRACE1:SPECTRUM:MEASview1:SELECT` selects Trace 1 to display the readout.

`TRACE1:SPECTRUM:MEASview1:SELECT?` might return `1`, indicating that the readout is being displayed for Trace 1.

TRACe<x>:TXGain

Sets or queries the display to show or hide the specified trace.

Conditions

Measurement view: Transmission Gain Where <x> is the trace and is 1, 2, or 3.

Group

Trace commands

Syntax

```
TRACe<x>:TXGain { OFF | ON | 0 | 1 }
TRACe<x>:TXGain?
```

Arguments

`OFF` or 0 hides the specified trace.

`ON` or 1 shows the specified trace.

Returns

`OFF` or 0 means that the specified trace is hidden.

`ON` or 1 means that the specified trace is showing.

Examples

`TRACE1:TXGAIN 1` shows Trace 1.

TRACe<x>:TXGain:AVERage:COUNt

Sets or queries the number of traces averaged to generate the displayed trace. (Present only when Function is set to anything but Normal.)

Conditions

Measurement view: Transmission Gain Where <x> is the trace and is 1, 2, or 3.

Group

Trace commands

Syntax

```
TRACe<x>:TXGain:AVERage:COUNt <number>
```

Arguments

`<number>::=<NR1>` specifies the number of traces to combine for averaging.

Range: 1 to 10000

Returns

`<NR1>` is an integer that is the number of traces to combine for averaging.

Examples

`TRACE1:TXGAIN:AVERAGE:COUNT 64` sets the average count to 64 for Trace 1.

TRACe<x>:TXGain:AVERage:RESet (No Query Form)

Clears/resets the average data and counter and restarts the averaging process.

Conditions

Measurement view: Transmission Gain Where <x> is the trace and is 1, 2, or 3.

Function must be is set to anything but Normal.

Group

Trace commands

Syntax

```
TRACe<x>:TXGain:AVERage:RESet
```

Arguments

None.

Examples

`TRACE1:TXGAIN:AVERAGE:RESET` clears average data and counter, and restarts the average process for Trace 1.

TRACe<x>:TXGain:COUNt

Sets or queries how many acquisitions run in the single acquisition mode for the Average or Max or Min Hold trace in the measurement.

Conditions

Measurement view: Transmission Gain Where <x> is the trace and is 1, 2, or 3.

Function must be is set to anything but Normal.

Group

Trace commands

Syntax

```
TRACe<x>:TXGain:COUNt <number>
TRACe<x>:TXGain:COUNt?
```

Arguments

<number>::=<NR1> specifies the count for Averaged, Max or Min Hold.

Range: 1 to 10000.

Returns

<NR1> is an integer that is the count for Averaged, Max or Min Hold.

Examples

TRACE1:TXGAIN:COUNT 32 sets the count to 32 for Trace 1.

TRACe<x>:TXGain:COUNt:ENABle

The command enables or disables whether to use the number of traces Averaged or Max Hold or Min Hold to generate the displayed trace. The query returns whether or not it is enabled.

Conditions

Measurement view: Transmission Gain Where <x> is the trace and is 1, 2, or 3.

Group

Trace commands

Syntax

```
TRACe<x>:TXGain:COUNt:ENABle { OFF | ON | 0 | 1 }
TRACe<x>:TXGain:COUNt:ENABle?
```

Arguments

OFF or 0 disables the count for the specified trace.

ON or 1 enables the count for the specified trace.

Returns

OFF or 0 means that the count for the specified trace is disabled.

ON or 1 means that the count for the specified trace is enabled.

Examples

TRACE1:TXGAIN:COUNT:ENABLE ON enables the count for Trace 1.

TRACe<x>:TXGain:COUNt:RESet (No Query Form)

Restarts the trace count when the function is Average, Max Hold, or Min Hold.

Conditions

Measurement view: Transmission Gain Where <x> is the trace and is 1, 2, or 3.

Group

Trace commands

Syntax

```
TRACe<x>:TXGain:COUNt:RESet
```

Arguments

None.

Examples

`TRACE1:TXGAIN:COUNT:RESET` clears the trace data and counter, and restarts the process for Trace 1.

TRACe<x>:TXGain:FREeze

Sets the trace as frozen or live. The query returns whether the trace is frozen or live.

Conditions

Measurement view: Transmission Gain Where <x> is the trace and is 1, 2, or 3.

Group

Trace commands

Syntax

```
TRACe<x>:TXGain:FREeze { OFF | ON | 0 | 1 }
TRACe<x>:TXGain:FREeze?
```

Arguments

`OFF` or 0 updates the display of the specified trace normally.

`ON` or 1 stops updating (freezes) the display of the specified trace.

Returns

`OFF` or 0 means that the specified trace is updating normally on the display.

`ON` or 1 means that the specified trace is not updating on the display.

Examples

`TRACE1:TXGAIN:COUNT:FREEZE ON` freezes Trace 1 in the display.

TRACe<x>:TXGain:FUNCtion

Sets or queries the trace processing method.

Conditions

Measurement view: Transmission Gain Where <x> is the trace and is 1, 2, or 3.

Group

Trace commands

Syntax

```
TRACe<x>:TXGain:FUNCtion { NORMal | MAXHold | MINHold | AVERage | AVGLog }
```

Arguments

`NORMal` selects the normal display.

`AVERage` selects the Average display that indicates the average signal level at each frequency point.

`MAXHold` selects the Max Hold display that indicates the maximum signal level at each frequency point.

`MINHold` selects the Min Hold display that indicates the minimum signal level at each frequency point.

`AVGLog` selects the Average of logs display that indicates the average signal level of logs at each frequency point.

Returns

See Arguments.

Examples

`TRACE1:TXGAIN:COUNT:FUNCTION MAXHold` selects Max Hold for Trace 1.

TRACe<x>:TXGain:SELect

Selects or queries the specified trace.

Conditions

Measurement view: Transmission Gain Where <x> is the trace and is 1, 2, or 3.

Group

Trace commands

Syntax

```
TRACe<x>:TXGain:SELect
TRACe<x>:TXGain:SELect?
```

Arguments

None.

Returns

The currently selected trace number.

Examples

`TRACE1:TXGAIN:SELECT` selects Trace 2.

`TRACE1:TXGAIN:SELECT?` might return `1`, indicating that Trace 1 is selected.

TRACe<x>:WLAN:AD:CONSte:RESUlt:CONTent:STATe

Sets or queries the state of the specified Result Contents (Preamble, Header, Data, Guard) of the specified trace.

Conditions

Measurement view: 802.11ad Constellation

Where the <x> trace variable is 1, 2 , 3, or 4

Group

Trace commands

Syntax

```
TRACe<x>:WLAN:AD:CONSte:RESUlt:CONTent:STATe <boolean>,<arg>
TRACe<x>:WLAN:AD:CONSte:RESUlt:CONTent:STATe?
```

Arguments

<boolean> arguments

0 or OFF disables the specified trace.

1 or ON enables the specified trace.

<arg> result contents arguments.

The Result Contents arguments are:

- 1 is the Preamble
- 2 is the Header
- 3 is the Data
- 4 is the Guard (only applies to Single Carrier standard)

Returns

See Arguments

Examples

TRACe1:WLAN:AD:CONSte:RESUlt:CONTent:STATe 1, 1 enables the Preamble result contents for trace 1.

TRACe2:WLAN:AD:CONSte:RESUlt:CONTent:STATe? might return 0, 3, indicating the Data result contents for trace 2 are disabled.

TRACe<x>:WLAN:AY:CONSte:RESUlt:CONTent:STATe

Set or query the result contents to Preamble, Header, Data, and Guard for the 11ay Single Carrier standard.

Conditions

Measurement view: 802.11ay Analysis Constellation

Group

Trace commands

Syntax

```
TRACe<x>:WLAN:AY:CONSte:RESUlt:CONTent:STATe { 0 | 1 }
TRACe<x>:WLAN:AY:CONSte:RESUlt:CONTent:STATe?
```

Arguments

<x> is 1,2,3,4

OFF or 0 disables the Preamble, Header, Data, and Guard result contents are disabled for the 11ay Single Carrier standard (for 4.32 GHz BW Header is not applicable) based on Trace <x> selection, where <x> is 1,2,3,4.

ON or 1 enables the Preamble, Header, Data, and Guard result contents are enabled for the 11ay Single Carrier standard (for 4.32 GHz BW Header is not applicable) based on Trace <x> selection, where <x> is 1,2,3,4.

Returns

OFF or 0 means that the Preamble, Header, Data, and Guard result contents are disabled for the 11ay Single Carrier standard (for 4.32 GHz BW Header is not applicable) based on Trace <x> selection, where <x> is 1,2,3,4.

ON or 1 means that the Preamble, Header, Data, and Guard result contents are enabled for the 11ay Single Carrier (for 4.32 GHz BW Header is not applicable) based on Trace <x> selection, where <x> is 1,2,3,4.

Examples

TRAC1:WLAN:AY:CONS:RESU:CONT:STAT ON enables the Preamble, Header, Data, and Guard result contents for the 11ay Single Carrier standard under the Trace 1 tab.

TRAC2:WLAN:AY:CONS:RESU:CONT:STAT? may return OFF, which means that the Preamble, Header, Data, and Guard result contents are disabled for the 11ay Single Carrier standard under the Trace 2 tab.

TRACe<x>:WLAN:CONSte

Turns the selected trace on or off in the constellation plot display, or queries the enabled status. < x> is accepted by the command but ignored as there is only a single WLAN constellation trace.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Trace commands

Syntax

```
TRACe<x>:WLAN:CONSte { 0 | 1 | OFF | ON }
TRACe<x>:WLAN:CONSte?
```

Arguments

OFF or 0 turns off the constellation trace.

ON or 1 turns on the constellation trace

Returns

Returns the status of the selected trace as an integral value.

0 implies a disabled trace.

1 implies a enabled trace.

Examples

TRAC:WLAN:CONS 1 turns the constellation trace on in the plot display.

TRACe<x>:WLAN:CONSte:FREeze

When the freeze state is turned on, the selected trace stops updating in the constellation plot. <x> is accepted by the command but ignored as there is only a single WLAN constellation trace.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Trace commands

Syntax

```
TRACe<x>:WLAN:CONSte:FREeze { 0 | 1 | OFF | ON }
TRACe<x>:WLAN:CONSte:FREeze?
```

Arguments

ON or 1 turns on the freeze state, causing the constellation trace to stop updating.

OFF or 0 turns off the freeze state.

Returns

Outputs the frozen status of the selected trace as an integral value. 1 indicates a frozen trace (stopped updating) and 0 a non-frozen trace.

Examples

TRAC:WLAN:CONS:FRE 1 freezes the constellation trace, causing it to stop updating.

TRACe<x>:WLAN:CRESponse:SELect

Sets or queries which WLAN Channel Response trace is the active trace.TRACe1 is Magnitude, TRACe2 is Phase.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Trace commands

Syntax

```
TRACe<x>:WLAN:CRESponse:SELect
TRACe<x>:WLAN:CRESponse:SELect?
```

Arguments

None

Returns

Returns 1, which implies that the selected trace is the active trace.

Returns 0, which implies that the selected trace is not the active trace.

Examples

TRACE1:WLAN:CRES:SELECT specifies to select the magnitude trace.

TRACE1:WLAN:CRES:SELECT? might return 1, indicating that the magnitude trace is the active trace.

TRACe<x>:WLAN:EVM:FREeze

When the freeze state is turned on, the selected trace stops updating. TRACe1 is Averages. TRACe2 is Points.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Trace commands

Syntax

```
TRACe<x>:WLAN:EVM:FREeze {0|1|OFF|ON}
```

TRACe<x>:WLAN:EVM:FREeze?

Arguments

ON or 1 turns on the freeze state, causing the selected trace to stop updating.

OFF or 0 turns off the freeze state.

Returns

Outputs the frozen status of the selected trace as an integral value. 1 indicates a frozen trace and 0 a non-frozen trace.

Examples

TRACE2:WLAN:EVM:FREEZE 1 causes Trace 2 to stop updating.

TRACE2:WLAN:EVM:FREEZE 1? might return 0, indicating that Trace 2 is not in a freeze state and is updating.

TRACe<x>:WLAN:EVM[:SHOW]

Displays or hides the selected trace. TRACe1 is Averages.TRACe2 is Points.

NOTE: Use this command only with multicarrier standards and not singlecarrier (802.11b).

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Trace commands

Syntax

```
TRACe<x>:WLAN:EVM[:SHOW] { 0 | 1 | OFF | ON}
TRACe<x>:WLAN:EVM[:SHOW]?
```

Arguments

ON or 1 displays the trace.

OFF or 0 hides the trace.

Examples

TRAC2:WLAN:EVM:SHOW 1 displays the WLAN EVM points trace.

TRACe<x>:WLAN:MERRor:FREeze

When the freeze state is turned on, the selected trace stops updating. TRACe1 is Averages. TRACe2 is Points.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Trace commands

Syntax

```
TRACe<x>:WLAN:MERRor:FREeze { 0 | 1 | OFF | ON }
TRACe<x>:WLAN:MERRor:FREeze?
```

Arguments

`ON` or 1 turns on the freeze state, causing the selected trace to stop updating.

`OFF` or 0 turns off the freeze state.

Returns

Outputs the frozen status of the selected trace as an integral value. 1 indicates a frozen trace and 0 a non-frozen trace.

Examples

`TRACE2:WLAN:MERR:FREEZE 1` causes Trace 2 to stop updating.

`TRACE2:WLAN:MERR:FREEZE 1?` might return `0`, indicating that Trace 2 is not in a freeze state and is updating.

TRACe<x>:WLAN:MERRor[:SHOW]

Displays or hides the selected trace. TRACe1 is Averages. TRACe2 is Points.

NOTE: Use this command only with multicarrier standards and not singlecarrier (802.11b).

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Trace commands

Syntax

```
TRACe<x>:WLAN:MERRor[:SHOW] { 0 | 1 | OFF | ON }
TRACe<x>:WLAN:MERRor[:SHOW]?
```

Arguments

`ON` or 1 displays the trace.

`OFF` or 0 hides the trace.

Examples

`TRAC2:WLAN:MERR:SHOW 1` displays the WLAN Magnitude Error points trace.

TRACe<x>:WLAN:PERRor:FREeze

When the freeze state is turned on, the selected trace stops updating. TRACe1 is Averages. TRACe2 is Points.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Trace commands

Syntax

```
TRACe<x>:WLAN:PERRor:FREeze { 0 | 1 | OFF | ON }
TRACe<x>:WLAN:PERRor:FREeze?
```

Arguments

`ON` or 1 turns on the freeze state, causing the selected trace to stop updating.

`OFF` or 0 turns off the freeze state.

Returns

Outputs the frozen status of the selected trace as an integral value. 1 indicates a frozen trace and 0 indicates a non-frozen trace.

Examples

`TRACE2:WLAN:PERR:FREEZE 1` causes Trace 2 to stop updating.

`TRACE2:WLAN:PERR:FREEZE 1?` might return `0`, indicating that Trace 2 is not in a freeze state and is updating.

TRACe<x>:WLAN:PERRor[:SHOW]

Displays or hides the selected trace. TRACe1 is Averages. TRACe2 is Points.

NOTE: Use this command only with multicarrier standards and not singlecarrier (802.11b).

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Trace commands

Syntax

```
TRACe<x>:WLAN:PERRor[:SHOW] { 0 | 1 | OFF | ON }
TRACe<x>:WLAN:PERRor[:SHOW]?
```

Arguments

`ON` or 1 displays the trace.

`OFF` or 0 hides the trace.

Examples

`TRAC2:WLAN:PERR:SHOW 1` displays the WLAN Phase Error points trace.

Trigger commands

*TRG (No Query Form)

Generates a trigger. It produces the same effect as the Force Trigger button on the Trigger control panel. This command is valid when the trigger mode is Triggered.

Conditions

Measurement views: All

Group

IEEE common commands

Syntax

`*TRG`

Arguments

None

Examples

`*TRG` generates a trigger.

TRIGger:ADVanced:SWEep:MODE

Sets or enables the analyzer to wait for a trigger for each acquisition in a swept spectrum or other measurement requiring multiple acquisitions worth of span.

Conditions

Requires an RSA be connected with SignalVu-PC.

Group

Trigger commands

```
Syntax
TRIGger:ADVanced:SWEep:MODE {OFF|ON|0|1}
```

Arguments

`ON` or 1 enables the wait for a trigger for each acquisition.

`OFF` or 0 disables the wait for a trigger for each acquisition.

Returns

Examples

`TRIGGER:ADVANCED:SWEEP:MODE 1` causes the analyzer to wait for a trigger for each acquisition.

TRIGger:DPX:SHOW:FRAMes

Determines when to show frames in the swept acquisition mode. Show frames continuously when ON or show only triggered frames when OFF.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger:DPX:SHOW:FRAMes { OFF | ON | 0 | 1 }
TRIGger:DPX:SHOW:FRAMes?
```

Arguments

`OFF` or 0 shows only triggered frames.

`ON` or 1 shows frames continuously.

Examples

`TRIGger:DPX:SHOW:FRAMes ON` shows frames continuously.

TRIGger:DPX:TDM:SHOW:ALL

Sets or queries the selections of the DPX Prefs control panel tab page.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger:DPX:TDM:SHOW:ALL { OFF | ON | 0 | 1 }
TRIGger:DPX:TDM:SHOW:ALL?
```

Arguments

`OFF` or 0 selects the Synch with acquisition triggers option.

`ON` or 1 selects the Trigger as fast as possible option.

Examples

`TRIGGER:DPX:TDM:SHOW:ALL ON` selects the Trigger as fast as possible option in the DPX Prefs control panel tab.

TRIGger:EVENt:EXTernal:SLOPe

Sets or queries the signal slope of the External input to determine a trigger event.

Conditions

Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group

Trigger commands

Syntax

```
TRIGger:EVENt:EXTernal:SLOPe { RISE | FALL }
```

Arguments

`RISE` Trigger event occurs on the rising edge.

`FALL` Trigger event occurs on the falling edge.

Examples

`TRIGGER:EVENT:EXTERNAL:SLOPE RISE` sets the trigger event to occur on the rising edge of the External signal input.

TRIGger:EVENt:INPut:LEVel

Sets or queries the power level that the signal must pass through to cause a trigger event when the trigger type is set to Power.

Conditions

Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group

Trigger commands

Syntax

`TRIGger:EVENt:INPut:LEVel <numeric>`

Arguments

`<numeric>::=<NR3>`

Range: –170 dBm to 50 dBm

Examples

`TRIGGER:EVENT:INPUT:LEVEL 10` sets the trigger threshold level to 10 dBm.

TRIGger:EVENt:INPut:SLOPe

Sets or queries the signal slope of the RF Input to determine a trigger event.

Conditions

Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group

Trigger commands

Syntax

`TRIGger:EVENt:INPut:SLOPe { RISE | FALL }`

Arguments

`RISE` Trigger event occurs on the rising edge.

`FALL` Trigger event occurs on the falling edge.

Examples

`TRIGGER:EVENT:INPUT:SLOPE RISE` sets the trigger event to occur on the rising edge of the RF Input signal.

TRIGger:EVENt:INPut:TYPE

Sets or queries the type of trigger. (Power is the only available setting.)

Conditions

Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group

Trigger commands

Syntax

```
TRIGger:EVENt:INPut:TYPE {POWer}
```

Arguments

`POWer` Sets trigger type to Power.

Examples

`TRIGGER:EVENT:INPUT:TYPE POWER` sets the trigger type to Power.

TRIGger:EVENt:SOURce

Sets or queries the source of the signal to be monitored for a trigger event.

Conditions

Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group

Trigger commands

Syntax

```
TRIGger:EVENt:SOURce { INPut | EXTernal }
TRIGger:EVENt:SOURce?
```

Arguments

`INPut` Sets the trigger source to the RF Input.

`EXTernal` Sets the trigger source to use the External input.

Examples

`TRIGGER:EVENT:SOURCE INPUT` sets the trigger source to the RF Input.

TRIGger:MASK:NEW (No Query Form)

Loads a new frequency mask.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger:MASK:NEW <freq(1)>,<ampl(1)>,<freq(2)>,<ampl(2)>,...,<freq(n)>,<ampl(n)>(n = 500 maximum)
```

Arguments

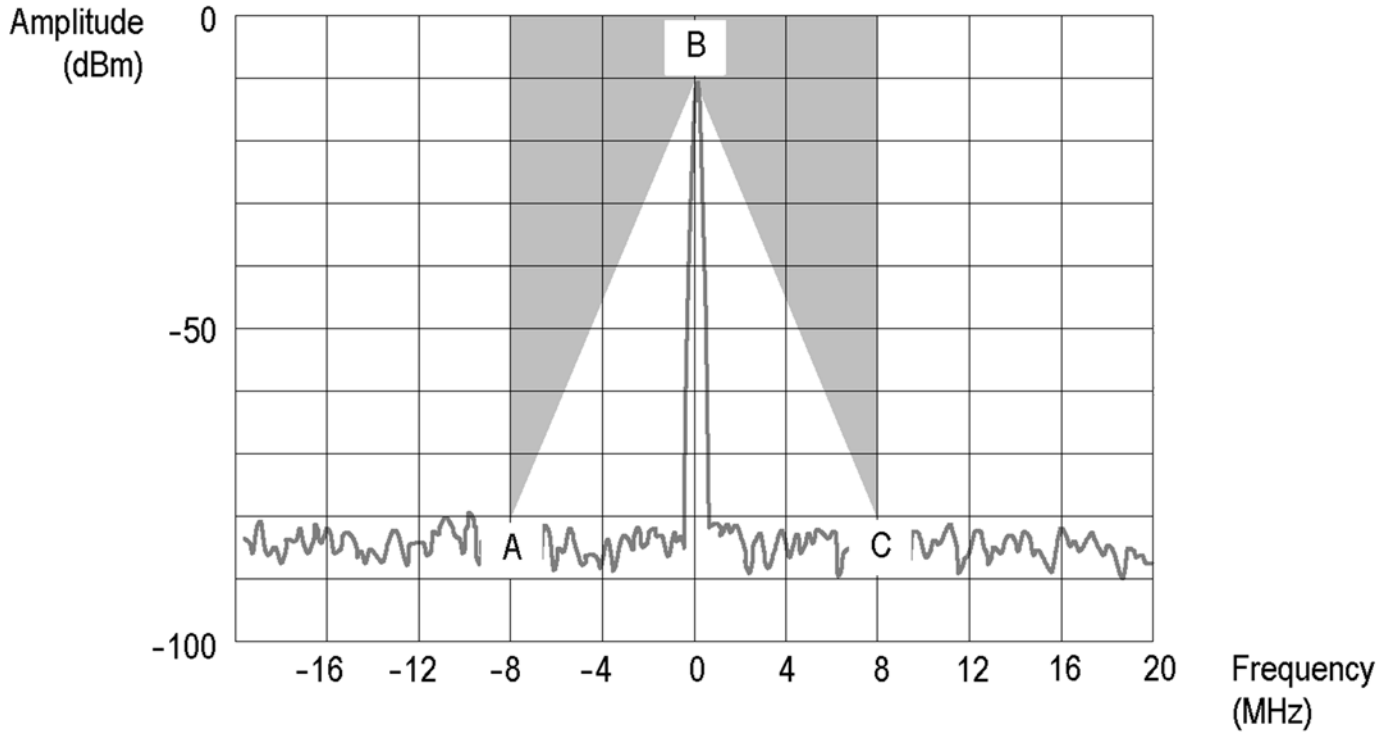
<freq(n)>,<ampl(n)> are numbers that represent a frequency (Hz) and amplitude (dBm) pair that define a point of the mask. Up to 500 pairs can be specified with zero frequency being the center screen. The mask is visible in the spectrum view with the following trigger conditions

NOTE: Do not include a unit suffix (such as Hz, dBm, or dB) with the arguments for this command.

- Trigger mode: Triggered (TRIGger[:SEQuence]:TIME:QUALified:TIME<x> is set to ON or 1.)
- Trigger type: Frequency Mask (TRIGger[:SEQuence]:EVENt:INPut:TYPE is set to FMASK.)

Examples

TRIGGER:MASK:NEW-8E6,-80,0,-10,8E6,-80 loads the mask with the points A (-8 MHz, -80 dBm), B (0 Hz, -10 dBm), and C (8 MHz, -80 dBm), as shown in the following figure.



Trigger mask setting example

TRIGger:MASK:NEW:AUTO (No Query Form)

Draws a new frequency mask automatically based on a reference trace.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

TRIGger:MASK:NEW:AUTO <meas_ID>,<trace_ID>,<x_margin>,<y_margin>

Arguments

<meas_ID>::<string> specifies the measurement view.

<trace_ID> specifies the reference trace.

The values of <meas_ID> and <trace_ID> are listed in the following table for each possible measurement view.

<x_margin>::<NRf> specifies the X margin (horizontal offset from the reference trace) in Hz.

<y_margin>::<NRf> specifies the Y margin (vertical offset from the reference trace) in dB.

Measurement view	<meas_ID>	<trace_ID>
Spectrum	"specan"	TRACE1 (Trace 1), TRACE2 (Trace 2), TRACE3 (Trace 3), TRACE4 (Math trace), TRACE5 (Spectrogram trace)
DPX, Spectrum	"dpx"	TRACE1 (+Peak trace), TRACE2 (-Peak trace), TRACE3 (Avg trace), TRACE4 (Math trace)q
GFFFGChannel power and ACPR	"acpr"	TRACE1 (Trace 1)
MCPR	"mcpr"	TRACE1 (Trace 1)
OBW	Q"obw"	TRACE1 (Trace 1)

Examples

TRIGGER:MASK:NEW:AUTO "specan",TRACE 1,2E+6,15 draws a new frequency mask automatically in the Spectrum view based on Trace 1 with the horizontal margin of 2 MHz and the vertical margin of 15 dB.

TRIGger:MASK:OPEN (No Query Form)

Opens a trigger mask with a specified file. To save a trigger mask, use the TRIGger:MASK:SAVE command.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

TRIGger:MASK:OPEN <file_name>

Arguments

<file_name>::<string> specifies the trigger mask file to open.

The file extension is .msk. You can omit the extension.

For the directory of file, refer to Specifying the File (on page 237).

Examples

TRIGGER:MASK:OPEN "C:\My Documents\Mask1" opens the trigger mask with the *Mask1* file in the *My Documents* directory.

TRIGger:MASK:SAVE (No Query Form)

Saves the current trigger mask to a specified file. To open the trigger mask, use the `TRIGger:MASK:OPEN` command.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger:MASK:SAVE <file_name>
```

Arguments

`<file_name>::<string>` specifies the file to save the trigger mask.

The file extension is .msk. You can omit the extension.

For the directory of file, refer to Specifying the File (on page 237).

Examples

`TRIGGER:MASK:SAVE"C:\My Documents\Mask1"` saves the trigger mask to the *Mask1* file in the *My Documents* directory.

TRIGger:SAVE:COUNt:MAXimum

Sets or queries the number of times a Save on Trigger action stores a file. After this limit is reached, no more files are saved. The instrument will continue to run, but no additional files are saved when a trigger event occurs.

Conditions

Requires an RSA be connected with SignalVu-PC.

Group

Trigger commands

Syntax

```
TRIGger:SAVE:COUNt:MAXimum <numeric>
```

Arguments

`<numeric>::=<NR1>`

Range: 1 to 10000

Examples

`TRIGGER:SAVE:COUNT:MAXIMUM 100` sets the number of files saved to 100.

TRIGger:SAVE:COUNt[:STATe]

Sets or queries the Maximum total number of files feature. When enabled and the set limit is reached, no more files are saved. The instrument will continue to run, but no additional files are saved when a trigger event occurs.

Conditions

Requires an RSA be connected with SignalVu-PC.

Group

Trigger commands

Syntax

```
TRIGger:SAVE:COUNt[:STATe] { OFF | ON | 0 | 1 }
```

Arguments

`ON` or 1 enables the maximum number of saved files feature.

`OFF` or 0 disables the maximum number of saved files feature.

Examples

`TRIGGER:SAVE:COUNT[:STATE] 1` enables the maximum number of saved files feature.

TRIGger:SAVE:DATA:FORMat

Sets or queries the file format used when saving the acquisition data on a trigger event.

Conditions

Requires an RSA be connected with SignalVu-PC.

Group

Trigger commands

Syntax

```
TRIGger:SAVE:DATA:FORMat { TIQ | CSV | MAT }
```

Arguments

`TIQ`: Native format data file which may include multiple frames.

`CSV`: Comma-separated values format.

`MAT`: MATLAB format.

Examples

`TRIGGER:SAVE:DATA:FORMAT TIQ` saves the data file in native format.

TRIGger:SAVE:DATA[:STATe]

Sets or queries to save data when a trigger event occurs.

Conditions

Requires an RSA be connected with SignalVu-PC.

Group

Trigger commands

Syntax

```
TRIGger:SAVE:DATA[:STATe] { OFF | ON | 0 | 1 }
```

Arguments

`ON` or 1 enables the save on trigger function.

OFF or 0 disables the save on trigger function

Examples

TRIGGER:SAVE:DATA[:STATE] 1 enables the save on trigger function.

TRIGger:SAVE:PICTure:FORMat

Sets or queries the file format used when saving a picture of the display when a trigger event occurs.

Conditions

Requires an RSA be connected with SignalVu-PC.

Group

Trigger commands

Syntax

TRIGger:SAVE:PICTure:FORMat { PNG | JPG | BMP }

Arguments

PNG, JPG, BMP: picture formats available.

Examples

TRIGGER:SAVE:PICTURE:FORMAT PNG sets the picture format extension to .PNG.

TRIGger:SAVE:PICTure[:STATe]

Sets or queries the save picture on trigger setting. This enables saving a picture of the display when a trigger event occurs.

Conditions

Requires an RSA be connected with SignalVu-PC.

Group

Trigger commands

Syntax

TRIGger:SAVE:PICTure[:STATe] { OFF | ON | 0 | 1 }

Arguments

OFF or 0 enables saving a picture.

ON or 1 disables saving a picture.

Examples

TRIGGER:SAVE:PICTURE[:STATE] 1 enables saving a picture of the display when a trigger event occurs.

TRIGger[:SEQuence]:ADVanced:HOLDoff

Sets or queries the trigger holdoff time which prevents triggers until there have been no trigger events for at least the specified holdoff period.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:ADVanced:HOLDoff <value>
TRIGger[:SEQuence]:ADVanced:HOLDoff?
```

Arguments

`<value>::<NRf>` specifies the trigger holdoff time. Range: 20 ns to 10 s.

Examples

`TRIGger[:SEQuence]:ADVanced:HOLDoff 1.5us` sets the trigger holdoff time to 1.5 µs.

TRIGger[:SEQuence]:ADVanced:HOLDoff:ENABle

Enables or queries the status of the Advanced Trigger Holdoff function.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:ADVanced:HOLDoff:ENABle { OFF | ON | 0 | 1 }
TRIGger[:SEQuence]:ADVanced:HOLDoff:ENABle?
```

Arguments

`OFF` or 0 does not trigger based on the advanced trigger holdoff settings.

`ON` or 1 enables triggering based on the advanced trigger holdoff settings.

Examples

`TRIGger:ADVanced:HOLDoff:ENABle ON` enables triggering based on the advanced trigger holdoff settings.

TRIGger[:SEQuence]:ADVanced:SWEep:MODE

Determines whether or not to trigger each segment in the swept acquisition mode.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:ADVanced:SWEep:MODE { OFF | ON | 0 | 1 }
TRIGger[:SEQuence]:ADVanced:SWEep:MODE?
```

Arguments

OFF or 0 does not trigger each segment in the swept acquisition mode.

ON or 1 triggers each segment in the swept acquisition mode.

Examples

TRIGGER:SEQUENCE:ADVANCED:SWEEP:MODE ON triggers each segment in the swept acquisition mode.

TRIGger[:SEQuence]:ADVanced:SWEPt:SEGMent:ENABle

Determines whether or not to set the spectrum analyzer to wait for a trigger for each acquisition in a swept spectrum or other measurement requiring multiple acquisitions worth of span.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:ADVanced:SWEPt:SEGMent:ENABle { OFF | ON| 0 | 1 }
TRIGger[:SEQuence]:ADVanced:SWEPt:SEGMent:ENABle?
```

Arguments

OFF or 0 does not trigger each segment in the swept acquisition mode.

ON or 1 triggers each segment in the swept acquisition mode.

Examples

TRIGger[:SEQuence]:ADVanced:SWEPt:SEGMent:ENABle ON waits for a trigger for each acquisition in a swept spectrum or other measurement requiring multiple acquisitions worth of span.

TRIGger[:SEQuence]:EVENt:A:EDGE:COUPling

Sets or queries the choice of trigger coupling type.

Conditions

Measurement views: All

Arguments

For MSO/LPD, available arguments are {DC | HFREject | LFREject | NOISereject}.

For SX, available arguments are {DC | AC | HFREject | LFREject | NOISereject| RF}.

Group

Trigger commands

Syntax

```
:TRIGger[:SEQuence]:EVENt:A:EDGE:COUPling {DC| AC | HFREject | LFREject | NOISereject| RF}
TRIGger[:SEQuence]:EVENt:A:EDGE:COUPling?
```

Examples

`TRIGger:SEQuence:EVENt:A:EDGE:COUPling DC` sets the Trigger coupling type to DC.

TRIGger[:SEQuence]:EVENt:A:EDGE:HOLDoff:MODE

Sets or queries the trigger hold off mode.

Conditions

Measurement views: All

This command is applicable only for SX/MSO.

Group

Trigger commands

Syntax

```
:TRIGger[:SEQuence]:EVENt:A:EDGE:HOLDoff:MODE {TIME | RANDom | AUTO}
:TRIGger[:SEQuence]:EVENt:A:EDGE:HOLDoff:MODE?
```

Examples

`:TRIGger:SEQuence:EVENt:A:EDGE:HOLDoff:MODE RANDom` sets Holdoff mode to Random.

TRIGger[:SEQuence]:EVENt:A:EDGE:HOLDoff:TIME

Sets or queries the time after triggering on an event, that the instrument waits before detecting the same trigger event to start the next acquisition.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:A:EDGE:HOLDoff:TIME <value>
TRIGger[:SEQuence]:EVENt:A:EDGE:HOLDoff:TIME?
```

Range

{0 to 10} in seconds

Examples

`TRIGger:SEQuence:EVENt:A:EDGE:HOLDoff:TIME 5` sets the hold off time to 5 seconds.

TRIGger[:SEQuence]:EVENt:A:EDGE:LEVEl

Sets or queries where the trigger point occurs for trigger source type.

Conditions

Measurement views: All

This cmd is applicable only for SX.

Group

Trigger commands

Syntax

```
:TRIGger[:SEQuence]:EVENt:A:EDGE:LEVEl <value>
:TRIGger[:SEQuence]:EVENt:A:EDGE:LEVEl
```

Examples

`:TRIGger:SEQuence:EVENt:A:EDGE:LEVEl -0.316` sets the trigger point to -0.316.

TRIGger[:SEQuence]:EVENt:EXTFront:IMPedance

Sets or queries the impedance of the external trigger input on the front panel.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:EXTFront:IMPedance { OHM50 | OHM5K}
TRIGger[:SEQuence]:EVENt:EXTFront:IMPedance?
```

Arguments

`OHM50` selects 50 Ω impedance.

`OHM5K` selects 5 kΩ impedance.

Examples

`TRIGGER:SEQUENCE:EVENT:EXTFRONT:IMPEDANCE OHM 50` selects 50 Ω impedance for the external trigger input on the front panel.

TRIGger[:SEQuence]:EVENt:EXTFront:LEVel

Sets or queries the trigger level at the external trigger input on the front panel.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:EXTFront:LEVel <value>
TRIGger[:SEQuence]:EVENt:EXTFront:LEVel?
```

Arguments

`<value>::<NRf>` specifies the trigger level. Range: -2.5 to +2.5 V.

Examples

`TRIGGER:SEQUENCE:EVENT:EXTFRONT:LEVEL 1.5` sets the trigger level to 1.5 V at the external trigger input on the front panel.

TRIGger[:SEQuence]:EVENt:EXTFront:SLOPe

Sets or queries the trigger slope of the external trigger input on the front panel.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:EXTFront:SLOPe { RISe | FALL }
TRIGger[:SEQuence]:EVENt:EXTFront:SLOPe?
```

Arguments

`RISe` causes the trigger event on the rising edge.

`FALL` causes the trigger event on the falling edge.

Examples

`TRIGGER:SEQUENCE:EVENT:EXTFRONT:SLOPE RISe` causes the trigger event on the rising edge of the external input signal.

TRIGger[:SEQuence]:EVENt:EXTRear:SLOPe

Sets or queries the trigger slope of the external trigger input on the rear panel. The trigger level is fixed to the TTL threshold.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:EXTRear:SLOPe { RISe | FALL }
TRIGger[:SEQuence]:EVENt:EXTRear:SLOPe?
```

Arguments

`RISe` causes the trigger event on the rising edge.

`FALL` causes the trigger event on the falling edge.

Examples

`TRIGGER:SEQUENCE:EVENT:EXTREAR:SLOPE RISe` causes the trigger event on the rising edge of the external input signal.

TRIGger[:SEQuence]:EVENt:GATed

Sets or queries the positive or negative logic for the gated trigger input on the rear panel.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:GATed { HIGh | LOW }
TRIGger[:SEQuence]:EVENt:GATed?
```

Arguments

`HIGh` specifies that the gated trigger input is high active.

`LOW` specifies that the gated trigger input is low active.

Examples

`TRIGGER:SEQUENCE:EVENT:GATED HIGh` specifies that the gated trigger input is high active.

TRIGger[:SEQuence]:EVENt:INPut:DDENsity:AMPLitude

Sets or queries the center vertical level for the density trigger region of the RF input DPX bitmap display.

Conditions

Measurement views: All

For SignalVu-PC users, the RSA instrument must be connected.

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:INPut:DDENsity:AMPLitude <value>
TRIGger[:SEQuence]:EVENt:INPut:DDENsity:AMPLitude?
```

Related Commands

TRIGger[:SEQuence]:EVENt:INPut:DDENsity:AMPLitude:TOLerance (on page 2719)

Arguments

`<value>::<NRf>` specifies the vertical center of the trigger region in dBm.

Range: 0 to 200 dBm.

Examples

`TRIGger[:SEQuence]:EVENt:INPut:DDENsity:AMPLitude 100dBm` sets the center level to 100 dBm for the density trigger region.

TRIGger[:SEQuence]:EVENt:INPut:DDENsity:AMPLitude:TOLerance

Sets or queries the positive and negative extensions of the density trigger region from the center vertical level. The density trigger region is defined for the DPX bitmap display.

Conditions

Measurement views: All

For SignalVu-PC users, the RSA instrument must be connected.

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:INPut:DDENsity:AMPLitude:TOLerance <value>
TRIGger[:SEQuence]:EVENt:INPut:DDENsity:AMPLitude:TOLerance?
```

Related Commands

TRIGger[:SEQuence]:EVENt:INPut:DDENsity:FREQuency:TOLerance (on page 2720)

Arguments

`<value>::<NRf>` specifies an absolute amplitude value for the positive and negative limits of the density trigger region in dBm. Range: within the defined measurement box.

Examples

`TRIGger[:SEQuence]:EVENt:INPut:DDENsity:AMPLitude:TOLerance 50` sets the positive limit 50 dBm above the amplitude level and the negative limit 50 dBm below the amplitude level.

TRIGger[:SEQuence]:EVENt:INPut:DDENsity:FREQuency

Sets or queries the center horizontal level for the density trigger region of the RF input DPX bitmap display.

Conditions

Measurement views: All

For SignalVu-PC users, the RSA instrument must be connected.

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:INPut:DDENsity:FREQuency <value>
TRIGger[:SEQuence]:EVENt:INPut:DDENsity:FREQuency?
```

Arguments

`<value>::<NRf>` specifies the horizontal center of the trigger region. Range: within the defined measurement box.

Examples

`TRIGger[:SEQuence]:EVENt:INPut:DDENsity:FREQuency 1.82GHz` sets the center horizontal level to 1.82 GHz for the density trigger region.

TRIGger[:SEQuence]:EVENt:INPut:DDENsity:FREQuency:TOLerance

Sets or queries the left and right extensions of the density trigger region from the center horizontal level. The density trigger region is defined for the DPX bitmap display.

Conditions

Measurement views: All

For SignalVu-PC users, the RSA instrument must be connected.

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:INPut:DDENsity:FREQuency:TOLerance <value>
TRIGger[:SEQuence]:EVENt:INPut:DDENsity:FREQuency:TOLerance?
```

Arguments

`<value>::<NRf>` specifies an absolute frequency value for the left and right limits from the center frequency of the density trigger region. Range: within the defined measurement box.

Examples

`TRIGger[:SEQuence]:EVENt:INPut:DDENsity:FREQuency:TOLerance 0.2GHz` for a center frequency of 1.82 GHz, sets the left horizontal limit to 1.62 GHz and the right horizontal limit to 2.02 GHz for the density trigger region.

TRIGger[:SEQuence]:EVENt:INPut:DDENsity:THReshold

Sets or queries the DPX Density threshold that defines a trigger event for the density trigger region of the DPX bitmap display. The DPX Density must be higher or lower (specified by the Density selection) than this value to define a trigger event.

Conditions

Measurement views: All

For SignalVu-PC users, the RSA instrument must be connected.

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:INPut:DDENsity:THReshold <value>
TRIGger[:SEQuence]:EVENt:INPut:DDENsity:THReshold?
```

Related Commands

Arguments

`<value>::<NRf>` specifies the DPX Density threshold as a percentage (%).

Range: within the defined measurement box.

Examples

`TRIGger[:SEQuence]:EVENt:INPut:DDENsity:THReshold 2` sets the DPX Density threshold to 2 % for the density trigger.

TRIGger[:SEQuence]:EVENt:INPut:DDENsity:VIOLation

Sets or queries whether a density value higher or lower than the THReshold value defines a trigger event for the density trigger region of the DPX bitmap display.

Conditions

Measurement views: All

For SignalVu-PC users, the RSA instrument must be connected.

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:INPut:DDENsity:VIOLation { HIGHer| LOWer }
TRIGger[:SEQuence]:EVENt:INPut:DDENsity:VIOLation?
```

Related Commands

TRIGger[:SEQuence]:EVENt:INPut:DDENsity:THReshold (on page 2721)

Arguments

`HIGHer` a DPX density value above the THReshold value defines the trigger event.

`LOWer` a DPX Density value below the THReshold value defines the trigger event.

Examples

`TRIGger:EVENt:INPut:DDENsity:VIOLation LOWER` sets the DPX Density violation to trigger when the probability of occurrence is lower than the threshold percentage.

TRIGger[:SEQuence]:EVENt:INPut:FEDGe:LEVel

Sets or queries the trigger level at the frequency edge input.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:INPut:FEDGe:LEVel <value>
TRIGger[:SEQuence]:EVENt:INPut:FEDGe:LEVel?
```

Arguments

`<value>::<NRf>` specifies the trigger level.

Range: ± (Max AcqBW/2) Hz

Examples

`TRIGGER:SEQUENCE:EVENT:INPUT:FEDGE:LEVEL 1.5` sets the trigger level to 1.5 V at the frequency edge input.

TRIGger[:SEQuence]:EVENt:INPut:FEDGe:LEVel:ACTual? (Query Only)

Queries the actual length of the frequency edge input for the DPX measurement.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:INPut:FEDGe:LEVel:ACTual?
```

Arguments

None

Returns

<NRf> The actual length in Hz.

Examples

TRIGGER:SEQUENCE:EVENT:INPUT:FEDGE:LEVEL:ACTUAL? might return 10 Hz, indicating that the actual length is 10 Hz.

TRIGger[:SEQuence]:EVENt:INPut:FEDGe:SLOPe

Sets or queries the trigger slope of the frequency edge input.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:INPut:FEDGe:SLOPe { RISe | FALL }
TRIGger[:SEQuence]:EVENt:INPut:FEDGe:SLOPe?
```

Arguments

RISe causes the trigger event on the rising edge.

FALL causes the trigger event on the falling edge.

Examples

TRIGGER:SEQUENCE:EVENT:INPUT:FEDGE:SLOPE RISE causes the trigger event on the rising edge of the frequency edge input.

TRIGger[:SEQuence]:EVENt:INPut:FEDGe:THReshold:LEVel

Set or queries the threshold level in the DPX measurement.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:INPut:FEDGe:THReshold:LEVel <value>
TRIGger[:SEQuence]:EVENt:INPut:FEDGe:THReshold:LEVel?
```

Arguments

<value>::<NRf> specifies the threshold level.

Range: 50 dBm to –170 dBm

Examples

TRIGGER:SEQUENCE:EVENT:INPUT:FEDGE:THRESHOLD:LEVEL –25 sets the threshold level to –25 dBm.

TRIGger[:SEQuence]:EVENt:INPut:FEDGe:THReshold:STATe

Set or queries the threshold state in the DPX measurement.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:INPut:FEDGe:THReshold:STATe { OFF | ON | 0 | 1 }
TRIGger[:SEQuence]:EVENt:INPut:FEDGe:THReshold:STATe?
```

Arguments

OFF or 0 disables the threshold.

ON or 1 enables the threshold.

Examples

TRIGGER:SEQUENCE:EVENT:INPUT:TDBWIDTH:STATE OFF enables the threshold.

TRIGger[:SEQuence]:EVENt:INPut:FMASk:{BANDwidth|BWIDth}[:RESolution]

Sets or queries the Resolution Bandwidth value to be used in the spectrum view for the frequency mask trigger.

Conditions

Measurement views: All

For SignalVu-PC users, the RSA instrument must be connected.

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:INPut:FMASk:{BANDwidth|BWIDth}[: RESolution] <value>
TRIGger[:SEQuence]:EVENt:INPut:FMASk:{BANDwidth|BWIDth}[:RESolution]?
```

Arguments

<value>::<NRf> specifies the resolution bandwidth for Frequency Mask triggering.

Examples

TRIGger[:SEQuence]:EVENt:INPut:FMASk:BANDwidth 300kHz sets the resolution bandwidth to 300 kHz.

TRIGger[:SEQuence]:EVENt:INPut:FMASk:BANDwidth|BWIDth[:RESolution]:ACTual? (Query Only)

Queries current setting for the Resolution Bandwidth value used in the spectrum view for the frequency mask trigger.

Conditions

Measurement views: All

For SignalVu-PC users, the RSA instrument must be connected.

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:INPut:FMASk:BANDwidth|BWIDth[: RESolution]:ACTual?
```

Arguments

None.

Examples

`TRIGger:EVENt:INPut:FMASk:BWIDth:ACTUAL?` returns the resolution bandwidth value.

TRIGger[:SEQuence]:EVENt:INPut:FMASk:BANDwidth|BWIDth[:RESolution]:AUTO

Sets or queries whether to automatically set the Resolution Bandwidth value used in the spectrum view for the frequency mask trigger.

Conditions

Measurement views: All

For SignalVu-PC users, the RSA instrument must be connected.

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:INPut:FMASk:BANDwidth|BWIDth[: RESolution]:AUTO { OFF | ON | 0 | 1 }
TRIGger[:SEQuence]:EVENt:INPut:FMASk:BANDwidth|BWIDth[: RESolution]:AUTO?
```

Arguments

`OFF` or `0` sets the resolution bandwidth manually with the command `TRIGger[:SEQuence]:EVENt:INPut:FMASk:{BANDwidth|BWIDth}[:RESolution]`.

`ON` or `1` sets the resolution bandwidth automatically.

Examples

`TRIGger:EVENt:INPut:FMASk:BWIDth:AUTO ON` sets the resolution bandwidth automatically.

TRIGger[:SEQuence]:EVENt:INPut:FMASk:VIOLation

Sets or queries when the analyzer triggers in the frequency mask trigger.

Conditions

Measurement views: All

For SignalVu-PC users, the RSA instrument must be connected.

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:INPut:FMASk:VIOLation { T | F | TF| FT | TFT | FTF }
TRIGger[:SEQuence]:EVENt:INPut:FMASk:VIOLation?
```

Arguments

The following table lists the arguments which represent the trigger violations.

Trigger violations

Violation	Description
T	Only one state is required to initiate a trigger event. The signal has at least one data point inside the mask. The trigger event occurs at the first point that appears inside the mask. A trigger event could occur on the first acquisition.
F	Only one state change is required to initiate a trigger event. The signal has at least one data point outside the mask. The trigger event occurs at the first point that appears outside the mask. A trigger event could occur on the first acquisition.
TF	Two states are required to initiate a trigger event. The signal must be inside the mask and then passes out of the mask. The trigger event occurs at the first transition where the signal passes out of the mask.
FT	Two states are required to initiate a trigger event. The signal must be outside the mask and then passes into the mask. The trigger event occurs at the first transition where the signal passes into the mask.
TFT	Three states are required to initiate a trigger event. The signal starts inside the mask and then passes out of the mask. Next, the signal must pass into the mask. The trigger event occurs at the second transition where the signal passes back into the mask.
FTF	Three states are required to initiate a trigger event. The signal starts outside the mask and then passes into the mask. Next, the signal must pass back outside the mask. The trigger event occurs at the second transition where the signal passes back out of the mask.

Examples

`TRIGGER:SEQUENCE:EVENT:INPUT:FMASK:VIOLATION TF` specifies that the analyzer will trigger when the signal has crossed into the mask and then outside of the mask.

TRIGger[:SEQuence]:EVENt:INPut:LEVel

Sets or queries the trigger level for the RF input level trigger.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:INPut:LEVel <value>
TRIGger[:SEQuence]:EVENt:INPut:LEVel?
```

Arguments

`<value>::<NRf>` specifies the trigger level. Range: -170 to +50 dBm.

Examples

`TRIGGER:SEQUENCE:EVENT:INPUT:LEVEL -10` sets the trigger level to −10 dBm for the RF input level trigger.

TRIGger[:SEQuence]:EVENt:INPut:RUNT:PULSe

Sets or queries the Runt trigger for a positive or negative going pulse.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:INPut:RUNT:PULSe { POSitive | NEGative }
TRIGger[:SEQuence]:EVENt:INPut:RUNT:PULSe?
```

Arguments

`POSitive` causes the trigger event on a positive runt pulse.

`NEGative` causes the trigger event on a negative runt pulse.

Examples

`TRIGger:EVENt:INPut:RUNT:PULSE POSitive` causes the trigger event on a positive pulse.

TRIGger[:SEQuence]:EVENt:INPut:RUNT:PULse:HIGH:LEVel

Sets or queries the upper level (first threshold) to qualify a Runt trigger.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:INPut:RUNT:PULse:HIGH:LEVel <value>
TRIGger[:SEQuence]:EVENt:INPut:RUNT:PULse:HIGH:LEVel?
```

Arguments

`<value>::<NRf>` sets the upper vertical level for a runt pulse in dBm.

Examples

`TRIGger:EVENt:INPut:RUNT:PULSE 300` causes the trigger event when the leading edge of a pulse crosses the 300 dBm level.

TRIGger[:SEQuence]:EVENt:INPut:RUNT:PULSe:LOW:LEVel

Sets or queries the lower level (second threshold) to qualify a Runt trigger.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:INPut:RUNT:PULSe:LOW:LEVel <value>
TRIGger[:SEQuence]:EVENt:INPut:RUNT:PULSe:LOW:LEVel?
```

Arguments

`<value>::<NRf>` sets the lower vertical level for a runt pulse in dBm.

Examples

`TRIGger:EVENt:INPut:RUNT:PULSE 3` causes the trigger event when the following edge of a pulse crosses the 3 dBm level.

TRIGger[:SEQuence]:EVENt:INPut:SLOPe

Sets or queries the trigger slope for the RF input level trigger.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:INPut:SLOPe { RISe | FALL }
TRIGger[:SEQuence]:EVENt:INPut:SLOPe?
```

Arguments

`RISe` causes the trigger event on the rising edge.

`FALL` causes the trigger event on the falling edge.

Examples

`TRIGGER:SEQUENCE:EVENT:INPUT:SLOPE RISe` causes the trigger event on the rising edge of the RF input signal.

TRIGger[:SEQuence]:EVENt:INPut:TDBWidth

Sets or queries the time-domain bandwidth for the RF input power trigger. This command is effective when `TRIGger[:SEQuence]:EVENt:INPut:TDBWidth:STATe` is ON.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:INPut:TDBWidth <value>
TRIGger[:SEQuence]:EVENt:INPut:TDBWidth?
```

Arguments

`<value>::<NRf>` specifies the time-domain bandwidth.

Range: 1 Hz to 60 MHz.

Examples

`TRIGGER:SEQUENCE:EVENT:INPUT:TDBWIDTH 5MHz` sets the time-domain bandwidth to 5 MHz for the RF input power trigger.

TRIGger[:SEQuence]:EVENt:INPut:TDBWidth:ACTual? (Query Only)

Queries the actual time-domain bandwidth for the RF input power trigger.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:INPut:TDBWidth:ACTual?
```

Related Commands

TRIGger[:SEQuence]:EVENt:INPut:TDBWidth (on page 2728)

Arguments

None

Returns

`<NRf>` Actual time-domain bandwidth.

Examples

`TRIGGER:SEQUENCE:EVENT:INPUT:TDBWIDTH:ACTUAL?` might return `1.000E+6`, indicating that the actual time-domain bandwidth is 1 MHz.

TRIGger[:SEQuence]:EVENt:INPut:TDBWidth:STATe

Determines whether to set the time-domain bandwidth automatically or manually for the RF input power trigger.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:INPut:TDBWidth:STATe { OFF | ON | 0 | 1 }
TRIGger[:SEQuence]:EVENt:INPut:TDBWidth:STATe?
```

Arguments

`OFF` or `0` sets the time-domain bandwidth automatically.

`ON` or `1` sets the time-domain bandwidth manually using the `TRIGger[:SEQuence]: EVENt:INPut:TDBWidth` command.

Examples

`TRIGGER:SEQUENCE:EVENT:INPUT:TDBWIDTH:STATE OFF` sets the time-domain bandwidth automatically.

TRIGger[:SEQuence]:EVENt:INPut:TYPE

Sets or queries the trigger type for the source of RF input.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:INPut:TYPE { POWer | FMASK | DDENsity | RUNT | FEDGe }
TRIGger[:SEQuence]:EVENt:INPut:TYPE?
```

Arguments

`POWer` uses the power level for triggering.

`FMASK` uses the frequency mask for triggering.

`DDENsity` uses the density mask for triggering.

`RUNT` uses the runt mask for triggering.

Examples

`TRIGGER:SEQUENCE:EVENT:INPUT:TYPE FMASK` uses the frequency mask for triggering.

TRIGger[:SEQuence]:EVENt:SOURce

Sets or queries the trigger event source.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:SOURce { INPut | EXTFront | EXTRear| EXTGated | LINe }
TRIGger[:SEQuence]:EVENt:SOURce?
```

Arguments

The following table lists the arguments.

Trigger event source

Argument	Source
INPut	RF input
EXTFront	Trigger in (front)
EXTRear	Trigger in (rear)
EXTGated	Gate in
LINe	AC line

Examples

`TRIGGER:SEQUENCE:EVENT:SOURCE INPut` specifies the trigger event source as the RF input.

TRIGger[:SEQuence]:FORCed

Determines whether or not to cause a manual trigger if the acquisition is armed, ready and waiting for a trigger. This command is valid when `TRIGger[:SEQuence]:TIME:QUALified:TIME<x>` is set to On (the trigger mode is Triggered).

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:FORCed { OFF | ON | 0 | 1 }
TRIGger[:SEQuence]:FORCed?
```

Arguments

`OFF` or `0` does not cause a manual trigger.

`ON` or `1` causes a manual trigger.

Examples

`TRIGGER:SEQUENCE:FORCED ON` causes a manual trigger if the acquisition is armed, ready and waiting for a trigger.

TRIGger:SEQuence:FORCed (No Query Form)

Causes a trigger event to occur when the trigger mode is set to Triggered, regardless of the set trigger conditions.

Conditions

Requires an RSA be connected with SignalVu-PC.

Group

Trigger commands

Syntax

```
TRIGger:SEQuence:FORCed
```

Arguments

None

Examples

`TRIGGER:SEQUENCE:FORCED` forces a trigger event.

TRIGger[:SEQuence]:IMMediate (No Query Form)

Causes a trigger immediately, skipping the event detection and delay. This command is valid when `TRIGger[:SEQuence]:TIME:QUALified:TIME<x>` is set to On (the trigger mode is Triggered).

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:IMMediate
```

Arguments

None

Examples

`TRIGGER:SEQUENCE:IMMEDIATE` causes a trigger immediately, skipping the event detection and delay.

TRIGger[:SEQuence]:STATus

Sets or queries the trigger mode (Free Run or Triggered).

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:STATus { OFF | ON | 0 | 1 }
TRIGger[:SEQuence]:STATus?
```

Arguments

`OFF` or `0` selects the free-run mode.

`ON` or `1` selects the triggered mode.

Examples

`TRIGGER:SEQUENCE:STATUS ON` selects the triggered mode.

TRIGger[:SEQuence][:STATus]

Sets or queries the trigger mode setting.

Conditions

Requires an RSA be connected with SignalVu-PC.

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence][:STATus] { 0 | 1 }
TRIGger[:SEQuence][:STATus]?
```

Arguments

`0` sets the trigger mode to Free Run.

`1` sets the trigger mode to Triggered.

Examples

`TRIGGER:SEQUENCE:STATUS 0` sets the trigger mode to Free Run.

TRIGger[:SEQuence]:TIME:DELay

Sets or queries the trigger delay time which occurs after recognizing an event but before actually declaring the trigger.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:TIME:DELay <value>
TRIGger[:SEQuence]:TIME:DELay?
```

Arguments

`<value>::<NRf>` specifies the trigger delay time. Range: 0 to 60 s.

Examples

`TRIGGER:SEQUENCE:TIME:DELAY 1.5` sets the trigger delay time to 1.5 s.

TRIGger[:SEQuence]:TIME:POSition

Sets or queries the position of the trigger event within the acquisition record.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:TIME:POSition <value>
TRIGger[:SEQuence]:TIME:POSition?
```

Arguments

`<value>::<NRf>` specifies the trigger position. Range: 1% to 99%.

Examples

`TRIGGER:SEQUENCE:TIME:POSITION 20` sets the trigger position to 20% of the acquisition record.

TRIGger[:Sequence]:TIME:POSition:AUTO

Sets or queries the position of the trigger event (automatic or manual) within the acquisition record.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:Sequence]:TIME:POSition:AUTO { OFF | ON | 0 | 1 }
TRIGger[:Sequence]:TIME:POSition:AUTO?
```

Arguments

`OFF` or `0` sets the position of the trigger event manually.

`ON` or `1` sets the position of the trigger event automatically.

Examples

`TRIGGER:SEQUENCE:TIME:POSITION:AUTO ON` sets the position of the trigger event automatically.

TRIGger[:SEQuence]:TIME:QUALified

Sets or queries the timing qualification setting for triggers. Trigger events that do not meet the timing qualifications are ignored.

Conditions

Measurement views: All

For SignalVu-PC users, the RSA instrument must be connected.

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:TIME:QUALified { SHORter | LONGer | INSide | OUTSide | NONE }
TRIGger[:SEQuence]:TIME:QUALified?
```

Arguments

SHORter the length of the trigger event is shorter than the time specified by the TRIGger:TIME:QUALified:TIME1 setting.

LONGer the length of the trigger event is longer than the time specified by the TRIGger:TIME:QUALified:TIME1 setting.

INSide the length of the trigger event is longer than the time TRIGger:TIME:QUALified:TIME1 setting and shorter than the TRIGger:TIME:QUALified:TIME2 setting.

OUTSide the length of the trigger event is shorter than the time TRIGger:TIME:QUALified:TIME1 setting or longer than the TRIGger:QUALified:TIME2 setting.

Examples

TRIGger:SEQuence:TIME:QUALified SHORter the length of the trigger event is shorter than the time specified by the TRIGger:TIME:DELAY setting.

TRIGger[:SEQuence]:TIME:QUALified:TIME<x>

Sets or queries the Time1 and Time2 parameters for time qualified triggering. Time1 is the shorter or first time, and Time2 is the second or later time for the qualifying period. Each TIME<x> must be set with a separate command.

Conditions

Measurement views: All

For SignalVu-PC users, the RSA instrument must be connected.

Group

Trigger commands

Syntax

TRIGger[:SEQuence]:TIME:QUALified:TIME<x> <value>

Related Commands

TRIGger[:SEQuence]:TIME:QUALified (on page 2734)

Arguments

<value>::<NRf> the length of the trigger time for TIME1 or TIME2.

(Range for RSA5100B: 0 ns to 28 s.)

Examples

TRIGger:TIME:QUALified:TIME1 55µs sets the TIME1 trigger point at 55 µs after the trigger event.

TRIGger[:SEQuence]:EVENt:A:EDGE:SETTings

Sets or queries the setting of trigger source type.

Conditions

Measurement views: All

This cmd is applicable only for SX.

Group

Trigger commands

Syntax

```
:TRIGger[:SEQuence]:EVENt:A:EDGE:SOURce:TYPE {SHARed|INDEpendent}
:TRIGger[:SEQuence]:EVENt:A:EDGE:SOURce:TYPE?
```

Arguments

`SHARed` specifies that all trigger settings will be shared among all trigger sources.

`INDEpendent` specifies that all trigger settings will be independent of trigger source.

Examples

`:TRIGger:SEQuence:EVENt:A:EDGE:SOURce SHARed` sets the source type settings to shared.

TRIGger[:SEQuence]:EVENt:A:EDGE:SLOPe

Sets or queries whether instrument finds the trigger point on rising or falling edge or either edge of a signal.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:A:EDGE:SLOPe { RISE | FALL | EITHer}
TRIGger[:SEQuence]:EVENt:A:EDGE:SLOPe?
```

Examples

`TRIGger:SEQuence:EVENt:A:EDGE:SLOPe RISE` sets the trigger point on Rise.

TRIGger[:SEQuence]:EVENt:A:EDGE:SOURce

Sets or queries the channel on which signal is to be monitored for a trigger event.

Conditions

Measurement views: All

For MSO/LPD trigger source can be selected from the list of available channels (CH1 to CH8).

For SX trigger source can be selected from list of channels as well as AUX and LINE inputs if available.

Group

Trigger commands

Syntax

```
:TRIGger[:SEQuence]:EVENt:A:EDGE:SOURce {CH1 | CH2 | CH3 | CH4 | CH5 | CH6 | CH7 | CH8 | MCH1 |
   MCH2 | MCH3 | MCH4 | AUX | LINE}
:TRIGger[:SEQuence]:EVENt:A:EDGE:SOURce?
```

Examples

`TRIGger:SEQuence:EVENt:A:EDGE:SOURce CH1` sets CH1 on which trigger event to be monitored.

TRIGger[:SEQuence]:EVENt:A:TYPE

Sets or queries the trigger type { supporting only Edge trigger}.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence:]EVENt:A:TYPE { EDGE }
TRIGger[:SEQuence:]EVENt:A:TYPE?
```

Examples

`TRIGger:SEQuence:EVENt:A:TYPE EDGE` sets trigger type to edge.

TRIGger[:SEQuence]:MODE

Sets or queries the trigger mode setting.

Conditions

This cmd is applicable only for SX.

Group

Trigger commands

Syntax

```
:TRIGger[:SEQuence]:MODE {0 | 1}
:TRIGger[:SEQuence]:MODE?
```

Examples

`:TRIGger:SEQuence:MODE 0` sets the trigger mode to Free Run.

TRIGger:TIME:POSition

Sets or queries the point (in percent) along the acquisition record the trigger event should occur. Manually setting the trigger position disables the Auto setting.

Conditions

Requires an RSA be connected with SignalVu-PC.

Group

Trigger commands

Syntax

```
TRIGger:TIME:POSition <numeric>
```

Arguments

`<numeric>::=<NR1>`

Range: 1 to 100

Examples

`TRIGGER:TIME:POSITION 50` sets the acquisition point to 50%.

TRIGger:TIME:POSition:AUTO

Sets or queries the instrument to automatically set the position of the trigger event.

Conditions

Requires an RSA be connected with SignalVu-PC.

Group

Trigger commands

Syntax

```
TRIGger:TIME:POSition:AUTO { OFF | ON | 0 | 1 }
```

Arguments

`ON` or `1` enables auto trigger position.

`OFF` or `0` disables auto trigger position.

Examples

`TRIGGER:TIME:POSITION:AUTO 1` enables the auto trigger position setting.

UNIT:POWer

Sets or queries the fundamental unit of power. This command is equivalent to `[SENSe]:POWer:UNITs`.

Conditions

Measurement views: All

Group

Unit commands

Syntax

```
UNIT:POWer { DBM | DBV | VOLTs | WATTs | DBUW | DBW| DBUV |DBMV | DBUA | DBUV_M | DBUA_M |
   AMPS }UNIT:POWer?
```

Arguments

The following table lists the arguments.

Power units

Argument	Power unit
DBM	dBm
DBV	dBV
VOLTs	Volts

Argument	Power unit
WATTs	Watts
DBUW	dBµW
DBW	dBW
DBUV	dBµV
DBMV	dBmV
DBUA	dBµA
DBUV_M	dBµV/m
DBUA_M	dBµA/m
AMPS	Amps

NOTE: Select dBµV/m or dBµA/m unit when using an antenna table.

Examples

`UNIT:POWERDBM` specifies the fundamental unit of power as dBm.

*WAI (No Query Form)

Prevents the analyzer from executing further commands or queries until all pending operations finish. This command allows you to synchronize the operation of the analyzer with your application program. For the details, refer to Synchronizing execution (on page 2748).

Conditions

Measurement views: All

Group

IEEE common commands

Syntax

`*WAI`

Related Commands

*OPC (on page 1544)

Arguments

None

Status and Events

The SCPI interface in the analyzer includes a status and event reporting system that enables the user to monitor crucial events that occur in the instrument. The analyzer is equipped with four registers and one queue that conform to IEEE Std 488.2-1987. This section will discuss these registers and queues along with status and event processing.

Status and event reporting system

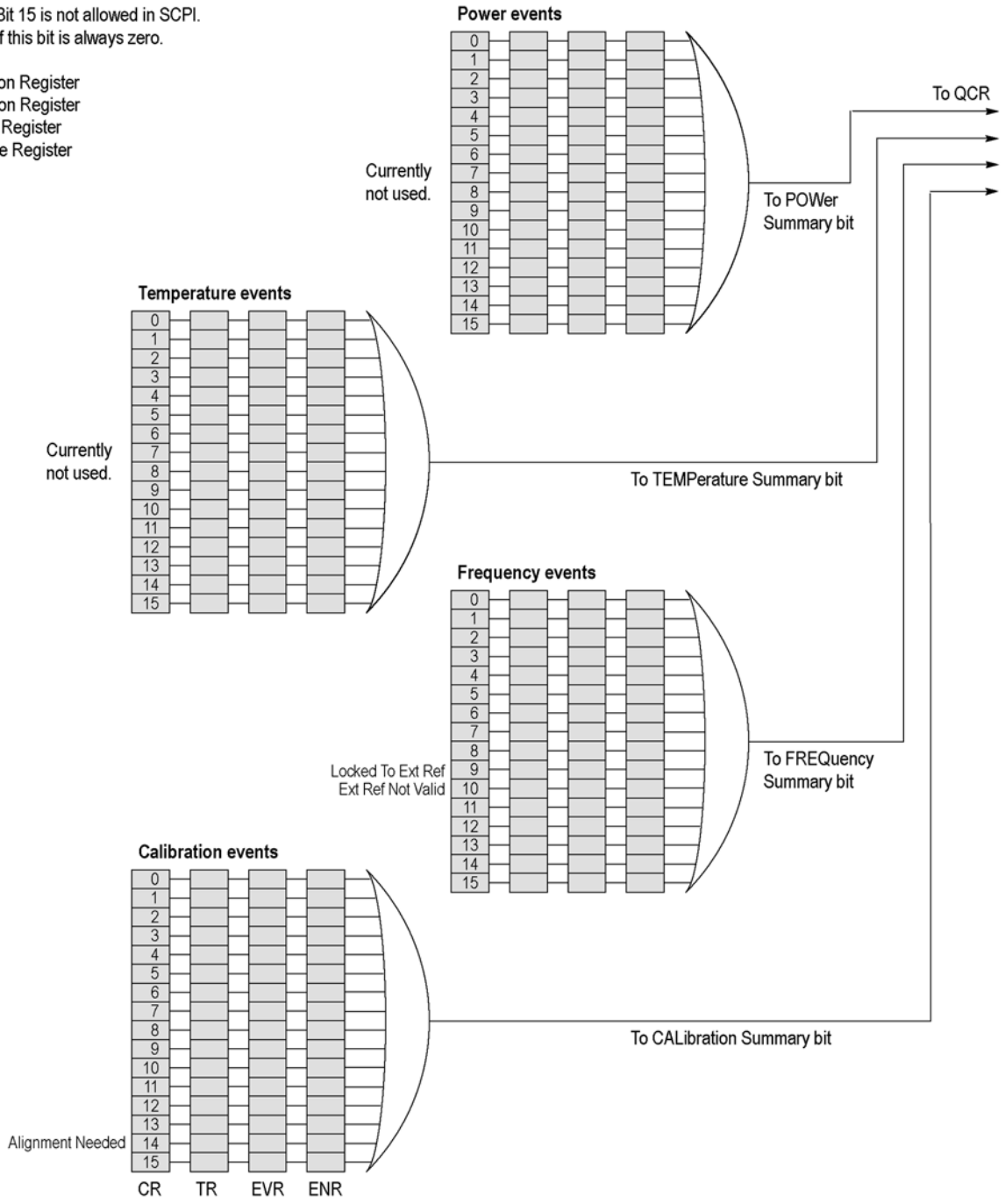
The following figure outlines the status and event reporting mechanism offered in the RSA Series analyzers. It contains three major blocks

- Standard Event Status
- Operation Status
- Questionable Status (fan-out structure)

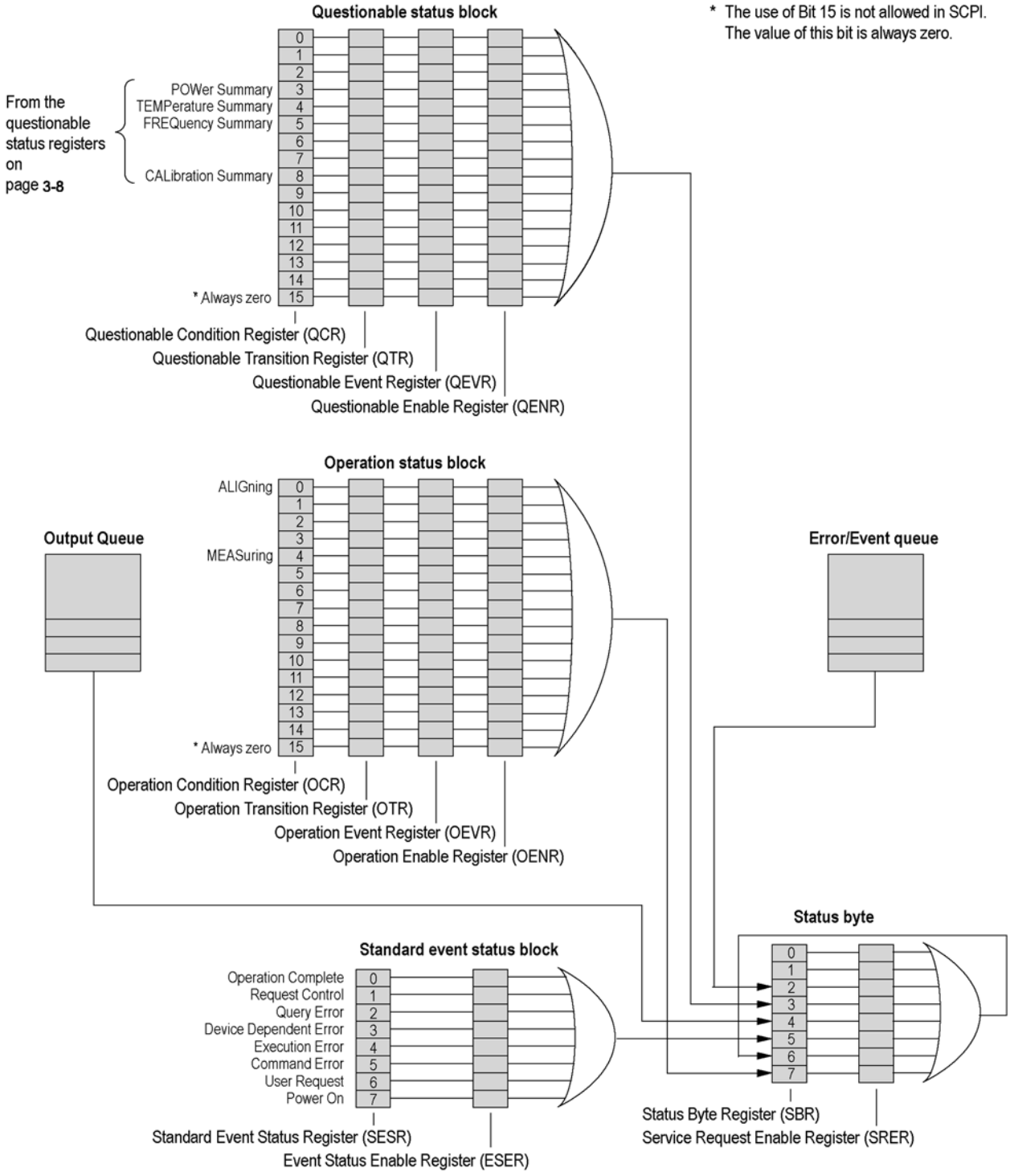
The processes performed in these blocks are summarized in the Status Byte. The three blocks contain four types of registers as shown in the following table.

Register type

Register	Description
Condition register	Records event occurrence in the instrument. Read only.
Transition register (positive/negative)	A positive transition filter allows an event to be reported when a condition changes from false to true. A negative filter allows an event to be reported when a condition changes from true to false. Setting both positive and negative filters true allows an event to be reported anytime the condition changes. Clearing both filters disables event reporting.
Event register	Records events filtered by the transition register. Read only.
Enable register	Masks the event register to report in the summary bit. User-definable.



Status/Event reporting mechanism



Status/Event reporting mechanism (continued)

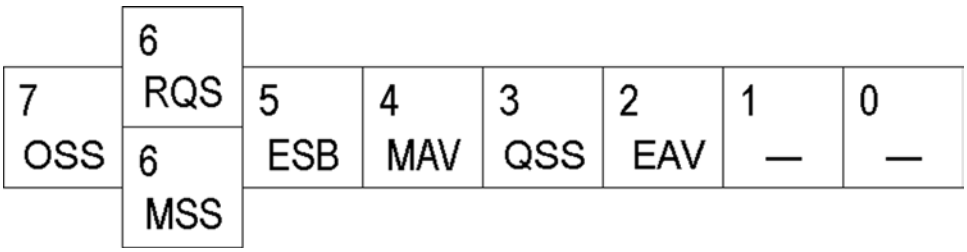
Status Byte

The Status Byte contains the following two registers

- Status Byte Register (SBR)
- Service Request Enable Register (SRER)

Status Byte Register (SBR)

The SBR is made up of 8 bits. Bits 4, 5 and 6 are defined in accordance with IEEE Std 488.2-1987. These bits are used to monitor the output queue, SESR and service requests, respectively. The contents of this register are returned when the *STB? query is used.



Status Byte Register (SBR)

SBR bit functions

Bit	Description
7	Operation Summary Status (OSS). Summary of the operation status register.
6	Request Service (RQS)/Master Status Summary (MSS). When the instrument is accessed using the GPIB serial poll command, this bit is called the Request Service (RQS) bit and indicates to the controller that a service request has occurred (in other words, that the GPIB bus SRQ line is LOW). The RQS bit is cleared when serial poll ends. When the instrument is accessed using the *STB? query, this bit is called the Master Status Summary (MSS) bit and indicates that the instrument has issued a service request for one or more reasons. The MSS bit is never cleared to 0 by the *STB? query.
5	Event Status Bit (ESB). This bit indicates whether or not a new event has occurred after the previous Standard Event Status Register (SESR) has been cleared or after an event readout has been performed.
4	Message Available Bit (MAV). This bit indicates that a message has been placed in the output queue and can be retrieved.
3	Questionable Summary Status (QSS). Summary of the Questionable Status Byte register.
2	Event Quantity Available (EAV). Summary of the Error Event Queue.
1-0	Not used

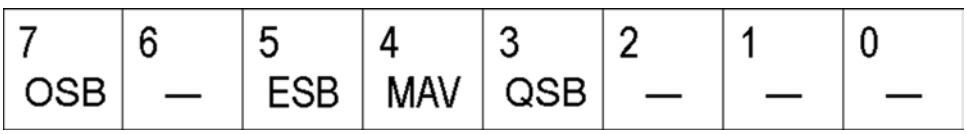
Service Request Enable Register (SRER)

The SRER is made up of bits defined exactly the same as bits 0 through 7 in the SBR as shown in the following figure. This register is used by the user to determine what events will generate service requests.

The SRER bit 6 cannot be set. Also, the RQS is not maskable.

The generation of a service request with the GPIB interface involves changing the SRQ line to LOW and making a service request to the controller. The result is that a status byte for which an RQS has been set is returned in response to serial polling by the controller.

Use the *SRE command to set the bits of the SRER. Use the *SRE? query to read the contents of the SRER. Bit 6 must normally be set to 0.



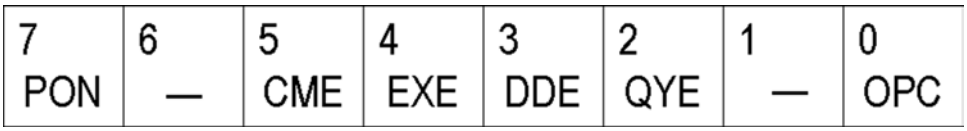
Service Request Enable Register (SRER)

Standard event status block

Reports the power on/off state, command errors, and the running state. It consists of the following registers of the following registers

- Standard Event Status Register (SESR)
- Event Status Enable Register (ESER)

These registers are made up of the same bits defined in the following figure and table. Use the *ESR? query to read the contents of the SESR. Use the *ESE() command to access the ESER.



Standard event status register

Standard event status register bit definition

Bit	Description
7	Power On (PON). Indicates that the power to the instrument is on.
6	Not used.
5	Command Error (CME). Indicates that a command error has occurred while parsing by the command parser was in progress.
4	Execution Error (EXE). Indicates that an error occurred during the execution of a command. Execution errors occur for one of the following reasons ▪ When a value designated in the argument is outside the allowable range of the instrument, or is in conflict with the capabilities of the instrument. ▪ When the command could not be executed properly because the conditions for execution differed from those essentially required.
3	Device-Dependent Error (DDE). An instrument error has been detected.
2	Query Error (QYE). Indicates that a query error has been detected by the output queue controller. Query errors occur for one of the following reasons ▪ An attempt was made to retrieve messages from the output queue, despite the fact that the output queue is empty or in pending status. ▪ The output queue messages have been cleared despite the fact that they have not been retrieved.
1	Not used.
0	Operation Complete (OPC). This bit is set with the results of the execution of the *OPC command. It indicates that all pending operations have been completed.

When an event occurs, the SESR bit corresponding to the event is set, resulting in the event being stacked in the Error/Event Queue. The SBR OAV bit is also set. If the bit corresponding to the event has also been set in the ESER, the SBR ESB bit is also set. When a message is sent to the Output Queue, the SBR MAV bit is set.

Operation status block

The operation status block contains conditions that are part of the instrument's normal operation. It consists of the following registers

- Operation Condition Register (OCR)
- Operation Positive/ Negative Transition Register (OPTR/ONTR)
- Operation Event Register (OEVR)
- Operation Enable Register (OENR)

These registers are made up of the same bits defined in the following table and figure. Use the STATus:OPERation commands to access the operation status register set.

15	14	13	12	11	10	9	8	7	6	5	4 MEAS	3	2	1	0 ALIG

Operation status register

Operation status register bit definition

Bit	Description
15	Always zero (0).
14 - 5	Not used.
4	Measuring (MEAS). Indicates that the instrument is actively measuring. When the measurement ends after this bit is set in measurement, it is reset. "In measurement" means that one of the following commands is in execution: ▪ INITiate commands ▪ READ commands
3 - 1	Not used.
0	Aligning (ALIG). Indicates that the instrument is currently performing an alignment. When the alignment ends after this bit is set in alignment, it is reset.

When the specified state changes in the OCR, its bit is set or reset. This change is filtered with a transition register, and the corresponding bit of the OEVR is set. If the bit corresponding to the event has also been set in the OENR, the SBR OSS bit is also set.

Questionable Status Block

The questionable status register set contains bits which give an indication of the quality of various aspects of the signal together with the fanned out registers as described in the next subsections. It consists of the following registers

- Questionable Condition Register (QCR)
- Questionable Positive/Negative Transition Register (QPTR/QNTR)
- Questionable Event Register (QEVR)
- Questionable Enable Register (QENR)

These registers are made up of the same bits defined in the following table and figure. Use the STATus:QUEStionable commands to access the questionable status register set.

15	14 CW	13	12	11	10	9	8 CAL	7	6	5 FREQ	4 TEMP	3 POW	2	1	0

Questionable status register

Questionable status register bit definition

Bit	Description
15	Always zero (0).
14	Command Warning (CW). Indicates a non-fatal warning that relates to the instrument's interpretation of a command, query, or one or more parameters of a specific command or query.
13 - 9	Not used.
8	CALibration Summary (CAL). Summary of the Questionable Calibration register.
7, 6	Not used.
5	FREQuency Summary (FREQ). Summary of the Questionable Frequency register.
4	TEMPerature Summary (TEMP). Summary of the Questionable Temperature register.
3	POWer Summary (POW). Summary of the Questionable Power register.
2 -0	Not used.

When the specified state changes in the QCR, its bit is set or reset. This change is filtered with a transition register, and the corresponding bit of the QEVR is set. If the bit corresponding to the event has also been set in the QENR, the SBR QSS bit is also set.

Questionable Power Register Set

Refines the power error for the POWer bit in the QCR.

Currently not used

Questionable Temperature Register Set

Refines the temperature error for the TEMPerature bit in the QCR.

Currently not used.

Questionable Frequency Register Set

The questionable frequency register set is made up of bits defined in the following table and figure. It refines the frequency error for the FREQuency bit in the QCR.

Use the STATus:QUEStionable:FREQuency commands to access the questionable frequency register set.

15	14	13	12	11	10 ERNV	9 LTER	8	7	6	5	4	3	2	1	0

Questionable frequency status register

Questionable frequency status register bit definition

Bit	Description
15	Always zero (0).

Bit	Description
14 - 11	Not used.
10	External Ref Not Valid (ERNV). Indicates that the external reference signal is not valid so the instrument is no longer locked to it.
9	Locked To External Ref (LTEF). Indicates that the instrument is locked to the external reference signal.
8 - 0	Not used.

Questionable Calibration Register Set

The questionable calibration register set is made up of bits defined in the following table and figure. It refines the calibration error for the CALibration bit in the QCR. Use the STATus:QUEStionable:CALibration commands to access the questionable calibration register set.



Questionable calibration status register

Questionable calibration status register bit definition

Bit	Description
15	Always zero.
14	Alignment Needed (ALN). Indicates the instrument needs the alignment.
13 - 0	Not used.

Queues

There are two types of queues in the status reporting system used in the analyzer: output queues and event queues.

Output Queue

The output queue is a FIFO (first in, first out) queue and holds response messages to queries, where they await retrieval. When there are messages in the queue, the SBR MAV bit is set.

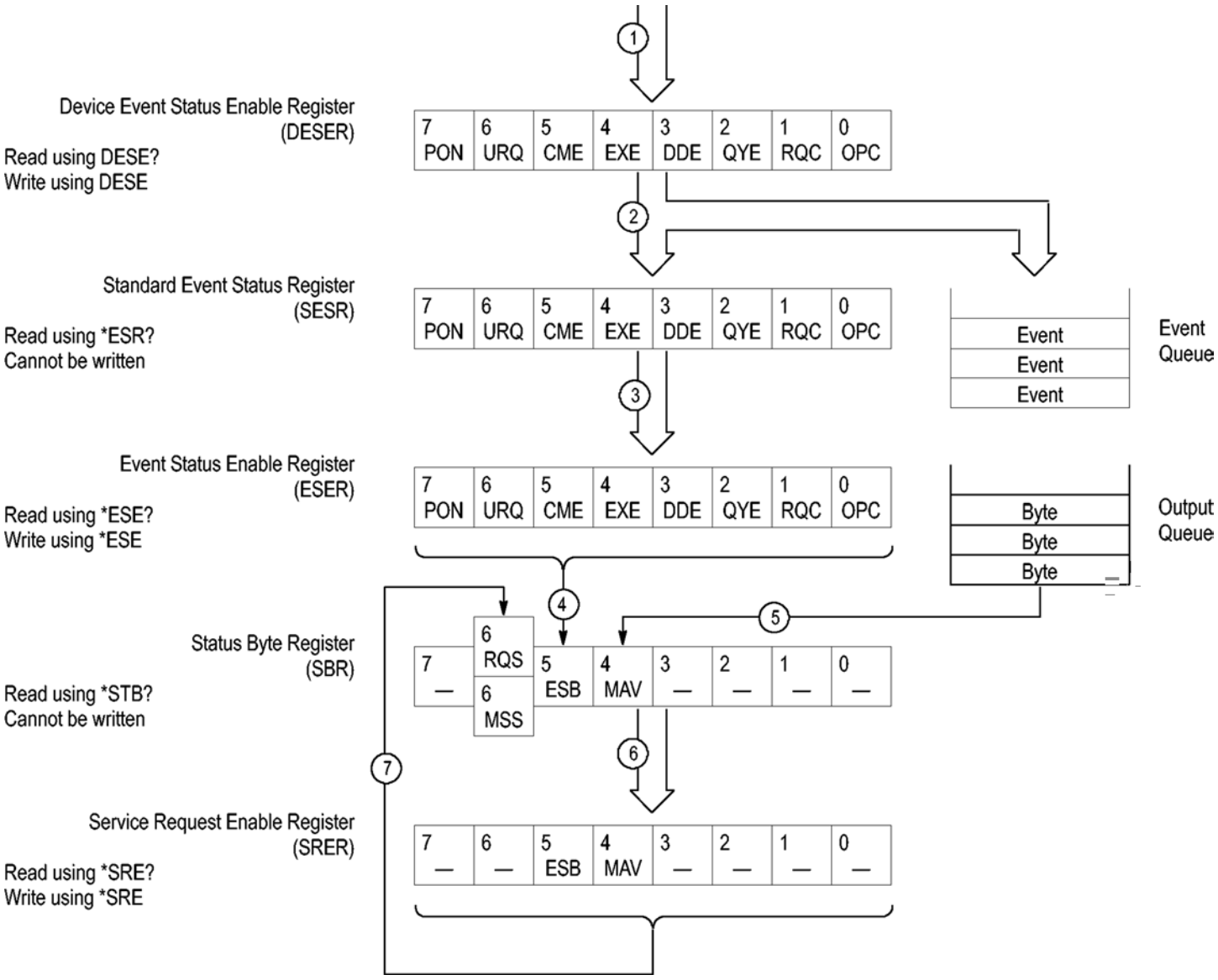
The output queue will be emptied each time a command or query is received, so the controller must read the output queue before the next command or query is issued. If this is not done, an error will occur and the output queue will be emptied; however, the operation will proceed even if an error occurs.

Event Queue

The event queue is a FIFO queue and stores events as they occur in the analyzer. If more than 32 events occur, event 32 will be replaced with event code -350 ("Queue Overflow"). The error code and text are retrieved using the SYSTem:ERRor queries.

Status and event processing sequence

The following figure shows an outline of the sequence for status and event processing.



Status and event processing sequence

1. If an event has occurred, the SESR bit corresponding to that event is set and the event is placed in the event queue.
2. A bit corresponding to that event in the ESER has is set.
3. The SBR ESB bit is set to reflect the status of the ESER.
4. When a message is sent to the output queue, the SBR MAV bit is set.
5. Setting either the ESB or MAV bits in the SBR sets the respective bit in the SRER.
6. When the SRER bit is set, the SBR MSS bit is set and a service request is generated when using the GPIB interface.

Synchronizing execution

Almost all commands are executed in the order in which they are sent from the controller, and the execution of each command is completed in a short period of time. However, the following commands perform data analysis in another thread, and another command can thus be executed concurrently

- `INITiate` commands
- `READ` commands
- `[SENSe]:REANalyze`

These commands are designed so that the next command to be sent is executed without waiting for the previous command to be completed. In some cases, a process executed by another command must first be completed before these commands can be executed; in other cases, these commands must be completed before the next command is executed.

You have two options to achieve command synchronization

- Using the status and event reporting function
- Using synchronizing commands

Using the Status and Event Reporting Function

In the following example, a READ command is used to obtain the measurement results while the Operation Condition Register (OCR) is being used to provide synchronization.

```
STATus:OPERation:NTRansition 16
// Set the filter of the OCR MEASuring bit
STATus:OPERation:ENABle 16
// Enable the filter of the OCR MEASuring bit
*SRE 128
// Set the SRER OSS bit
READ:SPECtrum:TRACe
// Obtain the measurement results
```

The command waits for generation of SRQ.

Using Synchronizing Commands

The IEEE-488.2 common commands include the following synchronizing commands

- *OPC
- *OPC?
- *WAI

Using the *OPC command. The *OPC command sets the SESR OPC bit when all the operations for which it is waiting are completed. If the GPIB interface is in use, you can synchronize the execution by using this command together with the serial polling or service request function.

The following is a command sequence example:

```
*ESE 1
// Enable the ESER OPC bit
*SRE 32
// Enable the SRER ESB bit
ABORt;INITiate:IMMediate;*OPC
// Wait for SRQ to provide synchronization
```

Using the *OPC? query. The query *OPC? writes ASCII code "1" into the Output Queue when all operations for which it is waiting are completed. You can provide synchronization using the command string as the following example:

```
ABORt;INITiate:IMMediate;*OPC
```

The command waits until "1" is written into the Output Queue. When the command goes to the Output Queue to read the data, a time-out may occur before the data is written into the queue.

Using the *WAI Command. After the process of the preceding command is completed, the *WAI command begins to execute the process of the next command as the following example:

```
ABORt;INITiate:IMMediate;*WAI
// Wait for the *WAI process to provide synchronization
```

Error messages and codes

Error codes with a negative value are SCPI standard error codes; errors with a positive value are unique to the RSA Series Signal Analyzers.

Event codes and messages can be obtained by using the queries `SYSTem:ERRor?` and `SYSTem:ERRor:ALL?` These are returned in the following format

```
<event_code>,"<event_message>"
```

Command errors

Command errors are returned when there is a syntax error in the command.

Command errors

Error code	Error message
-100	Command error
-101	Invalid character
-102	Syntax error
-103	Invalid separator
-104	Data type error
-105	GET not allowed
107	Can't replay - settings not compatible with data record
-109	Missing parameter
-111	Header separator error
-112	Program mnemonic too long
-113	Undefined header
-114	Header suffix out of range
-120	Numeric data error
-121	Character
-123	Exponent too large
-124	Too many digits
-128	Numeric data not allowed
-130	Suffix error

Error code	Error message
-131	Invalid suffix
-134	Suffix too long
-138	Suffix not allowed
-140	Character data error
-141	Invalid character data
-144	Character data too long
-148	Character data not allowed
-150	String data error
-151	Invalid string data
-158	String data not allowed
-160	Block data error
-161	Invalid block data
-168	Block data not allowed
-170	Command expression error
-171	Invalid expression
-178	Expression data not allowed
-180	Macro error
-181	Invalid outside macro definition
-183	Invalid inside macro definition
-184	Macro parameter error

Execution errors

These error codes are returned when an error is detected while a command is being executed.

Execution errors

Error code	Error message
-200	Execution error

Error code	Error message
-201	Invalid while in local
-202	Settings lost due to RTL
-210	Trigger error
-211	Trigger ignored
-212	Arm ignored
-213	Init ignored
-214	Trigger deadlock
-215	Arm deadlock
-220	Parameter error
-221	Settings conflict
-222	Data out of range
-223	Too much data
-224	Illegal parameter value
-225	Out of memory
-226	Lists not same length
-230	Data corrupt or stale
-231	Data questionable
-240	Hardware error
-241	Hardware missing
-250	Mass storage error
-251	Missing mass storage
-252	Missing media
-253	Corrupt media
-254	Media full
-255	Directory full

Error code	Error message
-256	Filename not found
-257	Filename error
-258	Media protected
-260	Execution expression error
-261	Math error in expression
-270	Execution macro error
-271	Macro syntax error
-272	Macro execution error
-273	Illegal macro label
-274	Execution macro parameter error
-275	Macro definition too long
-276	Macro recursion error
-277	Macro redefinition not allowed
-278	Macro header not found
-280	Program error
-281	Cannot create program
-282	Illegal program name
-283	Illegal variable name
-284	Program currently running
-285	Program syntax error
-286	Program runtime error

Device specific errors

These error codes are returned when an internal instrument error is detected. This type of error may indicate a hardware problem.

Device specific errors

Error code	Error message
-300	Device specific error
-310	System error
-311	Memory error
-312	PUD memory lost
-313	Calibration memory lost
-314	Save/Recall memory lost
-315	Configuration memory lost
-330	Self test failed
-350	Queue overflow

Query Errors

These error codes are returned in response to an unanswered query.

Query errors

Error code	Error message
-400	Query error
-410	Query interrupted
-420	Query unterminated
-430	Query deadlocked
-440	Query unterminated after indefinite period

Device errors

These error codes are unique to the RSA Series. They are classified into three groups: global, measurement, and source conditions, as shown in the following tables.

Device errors, global condition

Error code	Error message
100	Setup error
101	Disabled: data is from swept acquisition
102	Disabled: swept settings; Acquire data while display is selected

Error code	Error message
103	Acquisition bandwidth too small for current setup
104	Can't get acquisition data record
105	Can't open the requested display
106	Analysis failure
107	Analysis length was limited
108	Analysis length too small for current setup
109	No math trace: unmatched trace lengths
110	Analysis time was adjusted
111	Not enough samples for current setup
112	Can't replay. Data is from swept acquisition.
113	Can't replay. Live data needed for swept settings.
114	Recall error: setup not completely restored
115	Recall failure: problem with file or file contents
116	Save failure: file not saved
117	Unexpected software error. Please cycle power and try again.
118	Export failure: file not saved
119	Export failure: unable to open results file for export. File not saved.
120	Search condition for this result is already defined.
121	Search condition for this result was not found.
124	Load failed: <filename>
125	Store error: file not saved.
126	No Math trace: unmatched trace X range
127	Not enough memory for measurement
128	Incomplete analysis
129	Not enough samples for current setup

Error code	Error message
130	Mask creation error: <reason message>

Device errors, measurement condition

Error code	Error message
1000	TDBW actual (TDBW: Time Domain Bandwidth)
1002	RBW increased to
1003	RBW limited by acquisition bandwidth to
1004	RBW conflict. Increase span or analysis length
1011	Audio disabled: configuration problem
1012	Audio Demod disabled: swept acquisition
1013	Audio Demod disabled: trigger in use
1014	Audio disabled: IF band outside Acq BW
1015	Calibration error. See Windows Event Viewer for error detail.
OBW errors	
1016	Analysis failure: AcqBW < MeasBW + (5 x RBW)
1017	Analysis failure: AcqBW must be 10 kHz or more
1018	x dB BW > Meas BW
Pulse errors	
1019	AcqBW too low for current Chirp BW setting
1020	Impulse Response only valid for Chirp modulation
1023	Not enough memory for measurement
Other measurements	
1024	BW actual (limited by Acq BW)
1025	CISPR not available in FastFrame. Uncheck FastFrame in the Acquire panel.
1026	Analysis length must be in auto.
1027	Carrier not found

Error code	Error message
1029	CISPR accuracy limited by acq memory. Adjust RBW or freq range.
1030	CISPR: Acq BW too small for RBW. Try increasing span or freq range.
1031	Insufficient data for CISPR. Acquire while display is selected.
1032	VBW increased - Analysis Length too short
1033	VBW does not use full Analysis Length.
1034	Analysis time was adjusted\nReached sample limit of

Device errors, source condition

Error code	Error message
2008	Not calibrated
2009	FMT RBW limited by Acq BW to <XX>
2010	FMT RBW increased by Acq BW to <XX>
2011	Hardware failure - see Windows Event Viewer
2012	FMT disabled: Acq BW too small for current setup
2013	Hardware failure detected by diagnostics
2014	FMT disabled: RefLev too low for Volts/Watts units
2015	FMT disabled: settings conflict with selected measurement
2016	Preamp freq range exceeded
2024	RBW limited by Span to <XX>
2025	RBW increased by Span to <XX>
2028	External frequency reference signal not valid. Using internal reference.
2029	Unable to lock to external frequency reference. Using internal reference.
2033	Alignment error. Please run alignment again. If the problem persists, contact your Tektronix Service Center.
2034	Operational error: unable to complete operation. Please try again. If the problem persists, contact your Tektronix Service Center. See Windows Event Viewer for error detail.
2035	Hardware error: unable to configure hardware. Please try again. If the problem persists, contact your Tektronix Service Center. See Windows Event Viewer for error detail.

Error code	Error message
2036	Shutting down - internal temperature is too high. Check fans and airflow. If the problem persists, contact your Tektronix Service Center.
2037	Hardware error detected. To clear error, exit and restart the application. If the problem persists, contact your Tektronix Service Center.
2038	Disabled: RefLev too low\nfor Volts/Watts units
2042	Ext Corr > 20 dB pk-pk in acq segment
2044	Combined External Correction tables exceed the 60 dB peak-to-peak limit. External Correction tables were disabled. Please check table values and try again.
2045	Disabled: FastFrame doesn't support swept settings.
2046	Attenuator use currently exceeds the cautionary limit of 1200 changes per hour. Operation was stopped to protect against premature wear-out. The monitor function can be temporarily disabled in the Amplitude control panel or over the programmatic interface. Specified lifetime for an attenuator is 10,000,000 changes.
2047	Disabled: settings conflict with selected measurement
2048	Done saving — restart with Stop, then Run. The maximum number of files saved per run have been saved. To restart another FastSave acquisition, send INITIATE.

Status Conditions

Status conditions do not appear in the event queue, they are only returned by a `:STATus:<measurement>EVENts?` query.

Global status

Global status conditions can be returned with any `:STATus:<measurement>EVENts?` query.

Global status conditions

Status code	Status message
10107	Analysis Length was limited
10108	Analysis length too small for current setup
10110	Analysis time was adjusted
10122	Search result not enabled
10123	Search result no longer available

Measurement status

Measurement status conditions are unique to specific measurement groups.

Measurement status conditions

Status code	Status message
Pulse Measurements	
11001	Avg Tx only available in power units
11005	Pulse detection error
11006	Setup error: Define > Measurement Point is too late
11007	No pulses found
11008	No FFT (not all pulses have results)
11020	No FFT (not enough pulses)
11022	Analysis time was adjusted - reached time limit
11039	No sidelobe found
GP Digital Modulation	
11009	No burst detected
11034	Display not valid for selected modulation type
11035	Samples per symbol > 200. Decrease Measurement BW
11036	Too many samples per symbol. Decrease Measurement BW
Chan Power and ACPR	
11010	Measuring Noise
Other	
11028	CISPR: (progress message)
11037	Ext corrections enabled for traces
Settling Time	
11041	Signal not settled
11042	Settled duration too short
11043	First transition not found
11044	Meas Length limited by Analysis Length to xx.x xs

Status code	Status message
11045	Not enough samples: increase Meas BW or Meas Length
11046	Measurement Length exceeds limit of xxM points

Source status

Source status conditions can be returned with any `:STATus:<measurement>EVENts?` query.

Source status conditions

Status code	Status message
12000	Data acquired during RF ADC overrange
12005	Aligning
12006	Not aligned
12007	Unaligned data
12009	Data from uncalibrated instrument
12010	Dither: manual control
12012	Data acquired during hardware failure
12014	Data acquired during RF dig gain overflow
12015	Locking to ext freq ref signal...
12018	RF Attenuator: manual control
12019	Saving acquisition
12020	Recalling acquisition data
12022	Disabled: data is from swept acquisition
12023	Disabled: Freq Mask Trigger in use
12026	Acq Sampling params: manual control
12027	Swept: RF Trig invalid for most signals
12030	Data acquired during RF ADC overrange
12031	Data acquired during RF dig gain overflow
12039	Decrease RefLev or increase\nAtten to prevent data errors

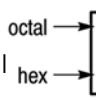
Status code	Status message
12040	Possible data errors - acquired with RefLev too much higher than RFAtten
12043	Ext Corr error - tables not applied
12056	Warming up
12057	Data acquired during warm-up
12063	Triggers disabled when DPX is swept
12064	Waiting for trigger. Open the DPX display.
12066	Data exceeds preamp range

Appendices

Appendix A: Character charts

B7 B6 B5 BITS B4 B3 B2 B1	0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
	CONTROL		NUMBERS SYMBOLS		UPPER CASE		LOWER CASE	
0 0 0 0	0 NUL 0 0	20 DLE 10 16	40 LA0 SP 20 32	60 LA16 0 30 48	100 TA0 @ 40 64	120 TA16 P 50 80	140 SA0 ` 60 96	160 SA16 p 70 112
0 0 0 1	1 GTL SOH 1 1	21 LL0 DC1 11 17	41 LA1 ! 21 33	61 LA17 1 31 49	101 TA1 A 41 65	121 TA17 Q 51 81	141 SA1 a 61 97	161 SA17 q 71 113
0 0 1 0	2 STX 2 2	22 DC2 12 18	42 LA2 " 22 34	62 LA18 2 32 50	102 TA2 B 42 66	122 TA18 R 52 82	142 SA2 b 62 98	162 SA18 r 72 114
0 0 1 1	3 ETX 3 3	23 DC3 13 19	43 LA3 # 23 35	63 LA19 3 33 51	103 TA3 C 43 67	123 TA19 S 53 83	143 SA3 c 63 99	163 SA19 s 73 115
0 1 0 0	4 SDC EOT 4 4	24 DCL DC4 14 20	44 LA4 $ 24 36	64 LA20 4 34 52	104 TA4 D 44 68	124 TA20 T 54 84	144 SA4 d 64 100	164 SA20 t 74 116
0 1 0 1	5 PPC ENQ 5 5	25 PPU NAK 15 21	45 LA5 % 25 37	65 LA21 5 35 53	105 TA5 E 45 69	125 TA21 U 55 85	145 SA5 e 65 101	165 SA21 u 75 117
0 1 1 0	6 ACK 6 6	26 SYN 16 22	46 LA6 & 26 38	66 LA22 6 36 54	106 TA6 F 46 70	126 TA22 V 56 86	146 SA6 f 66 102	166 SA22 v 76 118
0 1 1 1	7 BEL 7 7	27 ETB 17 23	47 LA7 ' 27 39	67 LA23 7 37 55	107 TA7 G 47 71	127 TA23 W 57 87	147 SA7 g 67 103	167 SA23 w 77 119
1 0 0 0	10 GET BS 8 8	30 SPE CAN 18 24	50 LA8 (28 40	70 LA24 8 38 56	110 TA8 H 48 72	130 TA24 X 58 88	150 SA8 h 68 104	170 SA24 x 78 120
1 0 0 1	11 TCT HT 9 9	31 SPD EM 19 25	51 LA9) 29 41	71 LA25 9 39 57	111 TA9 I 49 73	131 TA25 Y 59 89	151 SA9 i 69 105	171 SA25 y 79 121
1 0 1 0	12 LF A 10	32 SUB 1A 26	52 LA10 * 2A 42	72 LA26 : 3A 58	112 TA10 J 4A 74	132 TA26 Z 5A 90	152 SA10 j 6A 106	172 SA26 z 7A 122
1 0 1 1	13 VT B 11	33 ESC 1B 27	53 LA11 + 2B 43	73 LA27 ; 3B 59	113 TA11 K 4B 75	133 TA27 [5B 91	153 SA11 k 6B 107	173 SA27 { 7B 123
1 1 0 0	14 FF C 12	34 FS 1C 28	54 LA12 , 2C 44	74 LA28 < 3C 60	114 TA12 L 4C 76	134 TA28 \ 5C 92	154 SA12 l 6C 108	174 SA28 ¦ 7C 124
1 1 0 1	15 CR D 13	35 GS 1D 29	55 LA13 – 2D 45	75 LA29 = 3D 61	115 TA13 M 4D 77	135 TA29] 5D 93	155 SA13 m 6D 109	175 SA29 } 7D 125
1 1 1 0	16 SO E 14	36 RS 1E 30	56 LA14 . 2E 46	76 LA30 > 3E 62	116 TA14 N 4E 78	136 TA30 ^ 5E 94	156 SA14 n 6E 110	176 SA30 ~ 7E 126
1 1 1 1	17 SI F 15	37 US 1F 31	57 LA15 / 2F 47	77 UNL ? 3F 63	117 TA15 O 4F 79	137 UNT – 5F 95	157 SA15 o 6F 111	177 RUBOUT (DEL) 7F 127
	ADDRESSED COMMANDS	UNIVERSAL COMMANDS	LISTEN ADDRESSES		TALK ADDRESSES		SECONDARY ADDRESSES OR COMMANDS	

KEY



GPIB code (with ATN asserted)
ASCII character
decimal

Tektronix
REF: ANSI STD X3.4-1977
IEEE STD 488.1-1987
ISO STD 646-2973

Appendix B: SCPI conformance information

All commands for the RSA Series analyzers are based on SCPI Version 1999.0. The following table lists the commands that are defined in the SCPI 1999.0 Standard. The other commands not listed in the table are not defined in the SCPI 1999.0 Standard.

SCPI 1999.0-defined commands

Command group	Command				
IEEE command	*CAL				
	*CLS				
	*ESE				
	*ESR				
	*IDN				
	*OPC				
	*OPT				
	*RST				
	*SRE				
	*STB				
	*TRG				
	*WAI				
ABORt	:ABORt				
INITiate	:INITiate	:CONTinuous			
		[:IMMediate]			
STATus	:STATus	:OPERation	:CONDition?		
			:ENABLe		
			[:EVENt]?		
			:NTRansition		
			:PTRansition		
		:PRESet			

Command group	Command				
		:QUEStionable	:CONDition?		
			:ENABle		
			[:EVENt]?		
			:NTRansition		
			:PTRansition		
SYSTem	:SYSTem	:COMMunicate	:GPIB	[:SELF]	:ADDRess
		:DATE			
		:ERRor	:ALL?		
			:CODE	:ALL??	
				[:NEXT]?	
			:COUNt?		
			[:NEXT]?		
		:PREset			
		:TIME			
		:VERSion?			
:UNIT	:UNIT	:POWer			

Appendix C: Creating a DPX Frame Access Plugin

A DPX Frame Access plugin gives access to DPX frames (bitmap and traces) from within the RSA process. The frame is accessed directly from its storage in memory, and so this method provides faster access to this data than from the **ASCII** (GPIB) interface.

A DPX Frame Access plugin consists of a managed assembly (dll) that contains a class that implements the IDPXFrameAccess interface. See below:

```
/// <summary>
/// If an external assembly is found that contains a type
/// that implements this interface, then it will be given
/// access to DPX frame data synchronous with data
/// coming from the hardware.
/// <summary>
public interface IDPXFrameAccess : IDisposable
{
/// <summary>
/// A frame is available. It will get called for
/// every stitch.
```

```
/// <summary>
/// <param name="frame">valid only during the duration
/// of the function call </param>
void FrameAvailable(IDPXFrame frame);
}
```

The object that is passed into the FrameAvailable function is accessed via its IDPXFrame interface. See below:

```
/// <summary>
/// A DPX frame
///
/// <summary> public interface IDPXFrame
{
/// <summary>
/// Start frequency for traces and bitmap, Hz
/// <summary> double StartFrequency { get; }
/// <summary>
/// Stop frequency for traces and bitmap, Hz
/// </summary>
double StopFrequency { get; }
/// <summary>
/// Top value of graph - current units
/// </summary>
double TopOfGraph { get; }
/// <summary>
/// Bottom value of graph - current units
/// </summary>
double BottomOfGraph { get; }
/// <summary>
/// What type of frame?
/// </summary>
DPXFrameType FrameType { get; }
/// <summary>
/// This is true if the frame is completed.
/// It will be false if swept, and not all
/// of the intermediate stitches have been
/// completed. It will be true when all stitches
/// have been completed, or if real-time.
/// </summary>
bool FrameComplete { get; }
/// <summary>
/// Get dimensions for bitmap
/// </summary>
/// <param name="rowCount"></param>
/// <param name="columnCount"></param>
void GetDimensions(out int rowCount, out int
columnCount);
/// <summary>
/// <param name="x"></param>
void GetReferenceToBitmap(out ushort[] x);
/// <summary>
/// Get reference to array holding bitmap.
/// Do not change the values in this array
/// </summary>
/// <param name="x"></param>
void GetReferenceToBitmap(out float[] x);
/// <summary>
```

```
/// Get copy of Trace1. This will be the Max trace
/// </summary>
/// <param name="yValues">function will allocate
/// the array, as necessary</param>
void GetTrace1(ref float[] yValues);
/// <summary>
/// Get copy of Trace2. This will be the Min trace
/// </summary>
/// <param name="yValues">function will allocate
/// the array, as necessary</param>
void GetTrace2(ref float[] yValues);
/// <summary>
/// Get copy of Trace3. This will be the Avg trace
/// </summary>
/// <param name="yValues">function will allocate
/// the array, as necessary</param>
void GetTrace3(ref float[] yValues);
}
public enum DPXFrameType
{
/// <summary>
Standard,
/// </summary> Enhanced,
}
```

When the DPX measurement is running within the RSA application, the IDPXFrameAccess.FrameAvailable function is called once for every frame that becomes available from the DPX processing hardware. When the DPX measurement is operating in Real Time mode, FrameAvailable will be called for every Real Time frame. When the DPX measurement is operating in Swept mode, the total frequency span is covered by stitching together multiple real time spans at increasing center frequencies. In this case, FrameAvailable will be called for every stitch, with IDPXFrame.FrameComplete set to true on the last stitch.

The DPXFrameAccess assembly should not use the reference to the IDPXFrame object outside of the FrameAvailable function. It is only valid within the context of the FrameAvailable function call.

The creator of a DPXFrameAccess plugin should refer to the information in this programmer manual about the format of bitmap and trace data returned in the IDPXFrame object. Specifically, the FETCh:DPX:TRACe:xxx? queries are applicable. Traces that are not enabled are not computed and will not be available in the IDPXFrame.

Cautions about processing time within FrameAvailable function – in Real Time mode, DPX frames become available at a constant rate. As more time is taken in the FrameAvailable function, less CPU time is available for processing RF acquisitions (all the other measurements). If too much time is taken in the FrameAvailable function, frames may be missed and the update rate of other measurements may significantly slow down as they operate at lower priority than the DPX measurement.

To create a DPX Frame Access plugin, start with the example file attached to the PDF version of this manual. It has a class that implements IDPXFrameAccess and gives examples of accessing the various properties and methods of IDPXFrame.

NOTE: An example text file, DPXFrameAccessExample.cs, is attached to this PDF file. To access the attached file, open the PDF file and click the paperclip icon on the lower-left side of the document viewer.

Note that the build of this assembly will need to reference the MeasurementsShared.dll assembly in the C:\Program Files\Tektronix\XXXX\RSA\Client directory, where XXXX is either RSA5100B, or SignalVu, depending on the product..

Copy the DPX Frame Access assembly you have created to the C:\Program Files\Tektronix\XXXX\RSA\Plugins\DPX directory (you will need to create the directory). When the DPX measurement is started, it will discover the assembly and the FrameAvailable function will be called.

Glossary

Acronym	Description
ASCII	Acronym for the American Standard Code for Information Interchange. Controllers transmit commands to the instrument using ASCII character encoding.
ASK	Acronym for Amplitude Shift Keying. The process, or result of a process, in which the amplitude of the carrier is varied in accordance with the state of a digital input signal.
BNF (Backus-Naur Form)	A standard notation system for command syntax diagrams. The syntax diagrams in this manual use BNF notation.
Controller	A computer or other device that sends commands to and accepts responses from the analyzer.
EVM (Error Vector Magnitude)	The magnitude of an error of an actual signal relative to an ideal signal in a constellation display.
GPIB	Acronym for General Purpose Interface Bus, the common name for the communications interface system defined in IEEE Std 488.
IEEE	Acronym for the Institute for Electrical and Electronics Engineers.
IS95	Acronym for Interim Standard-95. The standards name for first-generation CDMA cellphone technology.
Modulation	The process of varying some characteristic of a signal with a second signal.
PSK	Acronym for Phase Shift Keying. The process, or result of a process, in which the carrier phase is varied discretely in accordance with a digital code.
QAM	Acronym for Quadrature Amplitude Modulation. The process, or result of a process, in which the amplitude and phase of the carrier are varied concurrently by synthesizing two orthogonal ASK waves (see ASK).