

**DSA8300 Series  
Digital Serial Analyzer  
Programmer Manual**



077-0570-00

**Tektronix**



**DSA8300 Series  
Digital Serial Analyzer  
Programmer Manual**

Copyright © Tektronix. 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.

IConnect and BERTScope are registered trademarks of Tektronix, Inc.

TekVISA is a trademark of Tektronix, Inc.

### **Contacting Tektronix**

Tektronix, Inc.  
14150 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.tektronix.com](http://www.tektronix.com) to find contacts in your area.

---

# Table of Contents

Preface .....	iii
Getting Started .....	1-1
Setting Up Remote Communications .....	1-3
Command Syntax .....	2-1
Command and Query Structure .....	2-1
Clearing the Instrument .....	2-3
Command Entry .....	2-4
Constructed Mnemonics .....	2-6
Argument Types .....	2-8
Command Groups .....	2-11
Acquisition Command Group .....	2-11
Calibration Command Group .....	2-12
Compensation Command Group .....	2-13
Cursor Command Group .....	2-15
Display Control Command Group .....	2-16
Hard Copy Command Group .....	2-17
Histogram Command Group .....	2-18
Horizontal Command Group .....	2-19
Mask Command Group .....	2-23
Math Command Group .....	2-25
Measurement Command Group .....	2-25
Miscellaneous Command Group .....	2-29
Phase Reference Command Group .....	2-29
Save and Recall Command Group .....	2-30
Status and Error Command Group .....	2-31
System Command Group .....	2-31
TDR Command Group .....	2-34
Trigger Command Group .....	2-35
Vertical Command Group .....	2-39
Waveform Database Command Group .....	2-41
Waveform Transfer Command Group .....	2-43
Commands Listed in Alphabetical Order .....	2-49
Status and Events .....	3-1
Registers .....	3-1
Queues .....	3-4
Event Handling Sequence .....	3-5
Synchronization Methods .....	3-6
Messages .....	3-11

Programming Examples .....	4-1
Appendix A: Character Set .....	A-1
Appendix B: Reserved Words .....	B-1
Appendix C: Factory Default Setup Values.....	C-1
Appendix D: GPIB Interface Specifications .....	D-1
Index	

---

# Preface

This programmer manual provides you with the information required to use GPIB commands for remotely controlling your instrument. This document supports the following instruments:

- DSA8300 Series





---

# Getting Started

This programmer manual provides you with the information required to use GPIB commands to remotely control your instrument. With this information, you can write computer programs that will perform functions such as setting the front-panel controls, taking measurements, performing statistical calculations, and exporting data for use in other programs, such as spreadsheets.

Besides the traditional GPIB electronic interface (referred to as the physical GPIB interface), your instrument has a *TekVISA*<sup>™</sup> GPIB-compatible interface (referred to as the virtual GPIB interface). This is a software Application Programming Interface (API) which enables you to communicate with the instrument in a variety of ways, including via the internet. With the following two exceptions, these interfaces are completely independent:

- **HEADER.** Command headers enabled or disabled on one interface are correspondingly enabled or disabled on the other interface. Refer to the command descriptions for more detailed information.
- **VERBOSE.** Verbosity enabled or disabled on one interface is correspondingly enabled or disabled on the other interface. Refer to the command description for more detailed information.

Most examples in this document assume that both **HEADER** and **VERBOSE** are **ON**.

The programmer manual is divided into the following major sections:

- **Getting Started.** This section introduces you to the programming information and provides basic information about setting up your instrument for remote control.
- **Syntax and Commands.** This section provides an overview of the command syntax that you use to communicate with the instrument and other general information about commands, such as how commands and queries are constructed, how to enter commands, constructed mnemonics, and argument types.
- **Command Groups.** This section contains all the commands listed by their functional groups. Each group consists of an overview of the commands in that group and a table that lists all the commands and queries for that group. You can click a command in the listing to display a detailed description of the command.
- **Commands Listed in Alphabetical Order.** This section contains all the commands in alphabetical order and is where you can find the complete description of each command.
- **Status and Events.** This section discusses the status and event reporting system for the GPIB interfaces. This system informs you of certain significant events that occur within the instrument. Topics that are discussed include registers, queues, event handling sequences, synchronization methods, and messages that the instrument may return, including error messages.
- **Appendices.** This section contains miscellaneous information, such as a list of reserved words, a table of the factory initialization (default) settings, and interface specifications that may be helpful when using commands to remotely control the instrument.

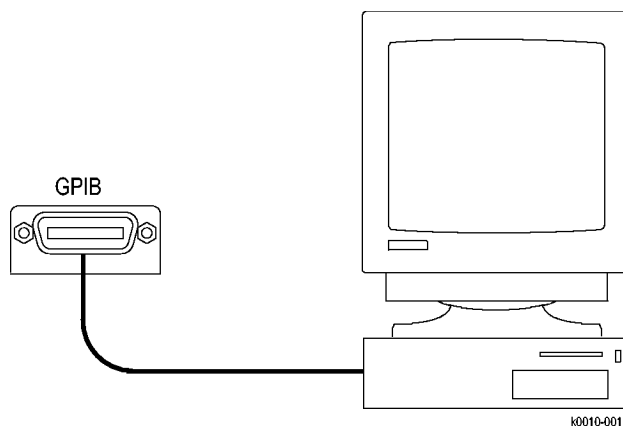
## Setting Up Remote Communications

Before setting up the instrument for remote communications using the electronic (physical) GPIB interface, you should familiarize yourself with the following GPIB requirements:

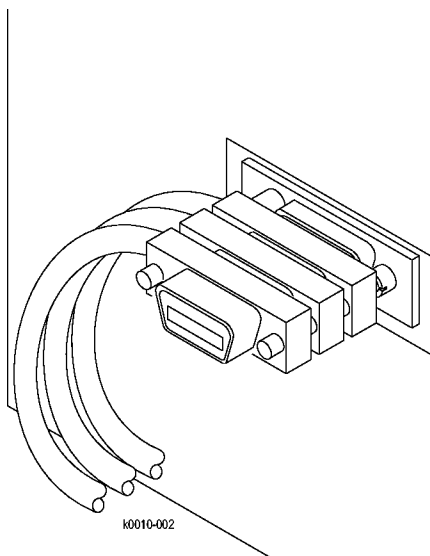
- A unique device address must be assigned to each device on the bus. No two devices can share the same device address.
- No more than 15 devices can be connected to any one line.
- One device should be connected for every 6 feet (2 meters) of cable used.
- No more than 65 feet (20 meters) of cable should be used to connect devices to a bus.
- At least two-thirds of the devices on the network should be powered on while using the network.
- Connect the devices on the network in a star or linear configuration. Do not use loop or parallel configurations.

### Connecting to the Instrument

Your instrument has a 24-pin GPIB connector on its rear (side) panel. This connector has a D-type shell and conforms to IEEE Std 488.1–1987. Attach an IEEE Std 488.1–1987 GPIB cable to this connector and to your controller as shown in the following figure.



If necessary, the GPIB connectors can be stacked as shown in the figure below.



### Setting the GPIB Address

To function correctly, your instrument must have a unique device address. The default settings for the GPIB configuration are:

- GPIB Address: 1
- GPIB Mode: GPIB Talk/Listen

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

1. Select **Utilities > User Preferences**.
2. Select the **GPIB Configuration** Tab.
3. Change the GPIB Address to a unique address.
4. Click **OK**.

The instrument is now set up for bidirectional communication with your controller.

### Talk/Listen Mode

The DSA8300 is factory set to Talk/Listen mode, which allows you to control the instrument via the GPIB port.

### Controller Mode

You can change the instrument to be a GPIB controller (master), which allows you to control other devices via the GPIB port.

### Off Bus Mode

In this mode, the instrument does not communicate with the GPIB. This mode is only available when the instrument is set to GPIB device mode.

---

**NOTE.** *Switching between Talk/Listen and Controller modes automatically reboots the instrument.*

---

# Command Syntax

You can control the operations and functions of the instrument through the GPIB interface using commands and queries. The related topics listed below describe the syntax of these commands and queries. The topics also describe the conventions that the instrument uses to process them. See the Command Groups topic in the table of contents for a listing of the commands by command group, or use the index to locate a specific command.

## Backus-Naur Form Notation

This documentation describes the commands and queries using Backus-Naur Form (BNF) notation. Refer to the following table for the symbols that are used.

**Table 2-1: Symbols for Backus-Naur Form**

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

## Command and Query Structure

Commands consist of set commands and query commands (usually called commands and queries). Commands modify instrument settings or tell the instrument to perform a specific action. Queries cause the instrument to return data and status information.

Most commands have both a set form and a query form. The query form of the command differs from the set form by its question mark on the end. For example, the set command `ACQUIre:MODE` has a query form `ACQUIre:MODE?`. Not all commands have both a set and a query form. Some commands have set only and some have query only.

## Messages

A command message is a command or query name followed by any information the instrument needs to execute the command or query. Command messages may contain five element types, defined in the following table.

**Table 2-2: Command Message Elements**

<b>Symbol</b>	<b>Meaning</b>
<Header>	This is the basic command name. If the header ends with a question mark, the command is a query. The header may begin with a colon (:) character. If the command is concatenated with other commands, the beginning colon is required. Never use the beginning colon with command headers beginning with a star (*).
<Mnemonic>	This is a header subfunction. Some command headers have only one mnemonic. If a command header has multiple mnemonics, a colon (:) character always separates them from each other.
<Argument>	This is a quantity, quality, restriction, or limit associated with the header. Some commands have no arguments while others have multiple arguments. A <space> separates arguments from the header. A <comma> separates arguments from each other.
<Comma>	A single comma is used between arguments of multiple-argument commands. Optionally, there may be white space characters before and after the comma.
<Space>	A white space character is used between a command header and the related argument. Optionally, a white space may consist of multiple white space characters.

**Commands**

Commands cause the instrument to perform a specific function or change one of the settings. Commands have the structure:

[ : ] <Header> [ <Space> <Argument> [ <Comma> <Argument> ] . . . ]

A command header consists of one or more mnemonics arranged in a hierarchical or tree structure. The first mnemonic is the base or root of the tree and each subsequent mnemonic is a level or branch off the previous one. Commands at a higher level in the tree may affect those at a lower level. The leading colon (:) always returns you to the base of the command tree.

**Queries** Queries cause the instrument to return status or setting information. Queries have the structure:

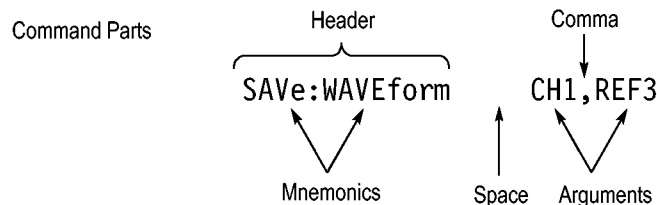
- [ : ] <Header> ?
- [ : ] <Header> ? [ <Space> <Argument> [ <Coma> <Argument> ] . . . ]

You can specify a query command at any level within the command tree unless otherwise noted. These branch queries return information about all the mnemonics below the specified branch or level. For example, HISTogram:STATistics:STDdev? returns the standard deviation of the histogram, while HISTogram:STATistics? returns all the histogram statistics, and HISTogram? returns all the histogram parameters.

**Headers** You can control whether the instrument returns headers as part of the query response. Use the HEADer command to control this feature. If header is on, the query response returns command headers, then formats itself as a valid set command. When header is off, the response includes only the values. This may make it easier to parse and extract the information from the response. The table below shows the difference in responses.

**Table 2-3: Comparison of Header Off and Header On Responses**

Query	Header Off	Header On
TIME?	"14:30:00"	:TIME"14:30:00"
ACQuire:NUMAVg?	100	:ACQUIRE:NUMAVG 100



## Clearing the Instrument

You can clear the Output Queue and reset the instrument to accept a new command or query by using the selected Device Clear (DCL) GPIB function. Refer to your GPIB library documentation for further details about the selected Device Clear operation.

## Command Entry

The following rules apply when entering commands:

- You can enter commands in upper or lower case.
- You can precede any command with white space characters. White space characters include any combination of the ASCII control characters 00 through 09 and 0B through 20 hexadecimal (0 through 9 and 11 through 32 decimal).
- The instrument ignores commands consisting of any combination of white space characters and line feeds.

### Abbreviating

You can abbreviate many instrument commands. Each command in this documentation shows the abbreviations in capitals. For example, you can enter the command ACQure:NUMAvg simply as ACQ:NUMA or acq:numa.

Abbreviation rules may change over time as new instrument models are introduced. Thus, for the most robust code, use the full spelling.

If you use the HEADer command to have command headers included as part of query responses, you can further control whether the returned headers are abbreviated or are full-length with the VERBose command.

### Concatenating

You can concatenate any combination of set commands and queries using a semicolon (;). The instrument executes concatenated commands in the order received.



When concatenating commands and queries, you must follow these rules:

- Separate completely different headers by a semicolon and by the beginning colon on all commands except the first one. For example, the commands `TRIGger:MODE EYE` and `ACQuire:NUMAVg 10`, can be concatenated into the following single command:

```
TRIGger:MODE EYE;:ACQuire:NUMAVg 10
```

- If concatenated commands have headers that differ by only the last mnemonic, you can abbreviate the second command and eliminate the beginning colon. For example, you can concatenate the commands `ACQuire:MODE ENVelope` and `ACQuire:NUMAVg 10` into a single command:

```
ACQuire:MODE ENVelope; NUMAVg 10
```

The longer version works equally well:

```
ACQuire:MODE ENVelope;:ACQuire:NUMAVg 10
```

- Never precede a star (\*) command with a colon:

```
ACQuire:MODE ENVelope;*OPC
```

Any commands that follow will be processed as if the star command was not there so the commands, `ACQuire:MODE ENVelope;*OPC;NUMAVg 10` will set the acquisition mode to envelope and set the number of acquisitions for averaging to 10.

- When you concatenate queries, the responses to all the queries are concatenated into a single response message. For example, if the TDR step state for channel 1 is on and it's polarity is negative, the concatenated query `:TDR:CH1:STEP:STATE?;POLARITY?` will return the following.

If the header is on:

```
:TDR:CH1:STEP:STATE 1 :TDR:CH1:STEP:POLARITY MINUS
```

If the header is off:

```
1;MINUS
```

- Set commands and queries may be concatenated in the same message. For example,

```
ACQuire:MODE SAMple;NUMAVg?;STATE?
```

is a valid message that sets the acquisition mode to sample. The message then queries the number of acquisitions for averaging and the acquisition state. Concatenated commands and queries are executed in the order received.

Here are some invalid concatenations:

```
DISPlay:STYle:NORMA1;ACQuire:NUMAVg 10 (no colon before ACQuire)
```

DISplay:COLor:CURSor1 1;:CURSor2 5 (extra colon before CURSor2; use DISplay:COLor:CURSor1 1;CURSor2 5 instead)

DISplay:STYle:NORMal;:\*OPC (colon before a star (\*) command)

DISplay:COLor:CURSor1 1;COLor:CURSor2 5 (levels of the mnemonics are different; either remove the second use of COLor or place :DISplay: in front of COLor:CURSor2 5)

**Terminating** This documentation uses <EOM> (End of message) to represent a message terminator.

**Table 2-4: End of Message Terminator**

Symbol	Meaning
<EOM>	Message terminator

The end-of-message terminator must be the END message (EOI asserted concurrently with the last data byte). The last data byte may be an ASCII linefeed (LF) character.

This instrument does not support ASCII LF only message termination. The instrument always terminates outgoing messages with LF and EOI. It allows white space before the terminator. For example, CR LF.

## Constructed Mnemonics

Some header mnemonics specify one of a range of mnemonics. For example, a channel mnemonic can be CH1, CH2, CH3, ... through CH8. You use these mnemonics in the command just as you do any other mnemonic. For example, there is a CH1:POSition command, and there is also a CH2:POSition command. In the command descriptions, this list of choices is abbreviated as CH<x>.

**Cursor Position Mnemonics** When cursors are displayed, commands may specify which cursor of the pair to use.

**Table 2-5: Cursor Mnemonics**

Symbol	Meaning
CURSOR<x>	A cursor selector; <x> is either 1 or 2.
POSITION<x>	A cursor selector; <x> is either 1 or 2.
HPOS<x>	A cursor selector; <x> is either 1 or 2.

### Histogram Statistics Specifier Mnemonics

Commands can specify which Sigma value to return for histogram statistics as a mnemonic in the header. A Sigma is specified in this way:

**Table 2-6: Histogram Statistics Specifier Mnemonics**

Symbol	Meaning
SIGMA<x>	A histogram statistics specifier; <x> is either 1, 2, or 3.

### Magnified Timebase Specifier Mnemonics

Commands can specify which of two magnified timebases to set or query as a mnemonic in the header. The magnified timebases are specified in this way:

**Table 2-7: Magnified Timebase Specifier Mnemonics**

Symbol	Meaning
MAG<x>	A magnified specifier; <x> is 1 or 2.

### Mask Specifier Mnemonics

Commands can specify which mask to set or query as a mnemonic in the header. The masks are specified in this way:

**Table 2-8: Mask Specifier Mnemonics**

Symbol	Meaning
MASK<x>	A mask specifier; <x> is 1 through 8.

### Measurement Specifier Mnemonics

Commands can specify which measurement to set or query as a mnemonic in the header. Up to eight automated measurements may be displayed with each displayed waveform. The displayed measurements are specified in this way:

**Table 2-9: Measurement Specifier Mnemonics**

Symbol	Meaning
MEAS<x>	A measurement specifier; <x> is 1 through 8.
SOURCE<x>	A waveform specifier; <x> is either 1 (Source 1 waveform) or 2 (Source 2 waveform).
REFLevel<x>	A waveform specifier for reference level measurements; <x> is either 1 (Source 1 waveform) or 2 (Source 2 waveform).
GATE<x>	A gate specifier; <x> is either 1 (Gate 1) or 2 (Gate 2).

### Channel Mnemonics

Commands specify the channel to use as a mnemonic in the header.

**Table 2-10: Channel Mnemonics**

Symbol	Meaning
CH<x>	A channel specifier; <x> is 1 through 8.

**Math Waveform Mnemonics**

Commands can specify the mathematical waveform to use as a mnemonic in the header.

**Table 2-11: Math Waveform Mnemonics**

Symbol	Meaning
Math<x>	A math waveform specifier; <x> is 1 through 8.

**Reference Waveform Mnemonics**

Commands can specify the reference waveform to use as a mnemonic in the header.

**Table 2-12: Reference Waveform Mnemonics**

Symbol	Meaning
REF<x>	A reference waveform specifier; <x> is 1 through 8.

**Waveform Database Mnemonics**

Commands can specify the reference waveform to use as a mnemonic in the header.

**Table 2-13: Waveform Database Mnemonics**

Symbol	Meaning
WFMDB<x>	A waveform database specifier; <x> is 1 through 4.

**Argument Types**

**Numeric**

Many instrument commands require numeric arguments. The syntax shows the format that the instrument returns in response to a query. This is also the preferred format when sending the command to the instrument though any of the formats will be accepted. This documentation represents these arguments as follows:

**Table 2-14: Numeric Arguments**

Symbol	Meaning
<NR1>	Signed integer value
<NR2>	Floating point value without an exponent
<NR3>	Floating point value with an exponent

Most numeric arguments will be automatically forced to a valid setting, either by rounding or truncating, when an invalid number is input unless otherwise noted in the command description.

**Quoted String**

Some commands accept or return data in the form of a quoted string, which is simply a group of ASCII characters enclosed by a single quote (') or double quote

("). The following is an example of a quoted string: "This is a quoted string". This documentation represents these arguments as follows:

**Table 2-15: Quoted String Argument**

Symbol	Meaning
<QString>	Quoted string of ASCII text

A quoted string can include any character defined in the 7-bit ASCII character set. Follow these rules when you use quoted strings:

1. Use the same type of quote character to open and close the string. For example: "this is a valid string".
2. You can mix quotation marks within a string as long as you follow the previous rule. For example, "this is an 'acceptable' string".
3. You can include a quote character within a string by repeating the quote. For example: "here is a "" mark".
4. Strings can have upper or lower case characters.
5. If you use a GPIB network, you cannot terminate a quoted string with the END message before the closing delimiter.
6. A carriage return or line feed embedded in a quoted string does not terminate the string, but is treated as just another character in the string.
7. The maximum length of a quoted string returned from a query is 1000 characters.

Here are some invalid strings:

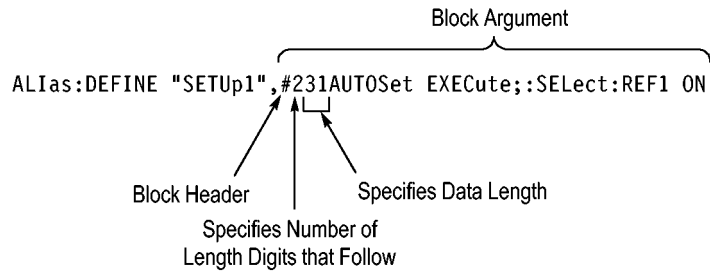
- "Invalid string argument" (quotes are not of the same type)
- "test<EOI>" (termination character is embedded in the string)

**Block** Several instrument commands use a block argument form (see the following table).

**Table 2-16: Block Argument**

Symbol	Meaning
<NZDig>	A nonzero digit character in the range of 1–9
<Dig>	A digit character, in the range of 0–9
<DChar>	A character with the hexadecimal equivalent of 00 through FF (0 through 255 decimal)
<Block>	A block of data bytes defined as: <Block> ::= {#<NZDig><Dig>[<Dig>...][<DChar>...]}  #0[<DChar>...]<terminator>}

<NZDig> specifies the number of <Dig> elements that follow. Taken together, the <NZDig> and <Dig> elements form a decimal integer that specifies how many <DChar> elements follow.



# Command Groups

The DSA8300 programmable interface conforms to Tektronix standard codes and formats except where noted. The GPIB interface also conforms to IEEE Std 488.2-1987 except where noted.

## Acquisition Command Group

Use the commands in the Acquisition Command Group to set up the modes and functions that control how the instrument acquires the signals you input to the channels and processes them into waveforms.

Using these commands for acquiring waveforms, you can do the following:

- Start and stop acquisitions.
- Control whether all waveforms are simply acquired, averaged, or enveloped over successive acquisitions.
- Set the controls or conditions that start and stop acquisitions.
- Determine the action the system takes upon completing an acquisition, such as saving all waveforms or saving the current display to a file when the acquisition is stopped.
- Get data on acquired waveforms, histograms, and masks.
- Get acquisition parameters.
- Clear all acquired data.

**Table 2-17: Acquisition Commands**

<b>Command</b>	<b>Description</b>
<a href="#">ACQUIRE?</a>	Returns acquisition parameters
<a href="#">ACQUIRE:CURRENTCOUNT:ACQWFMS?</a>	Returns acquired waveforms count
<a href="#">ACQUIRE:CURRENTCOUNT:HISTHITS?</a>	Returns histogram hits count
<a href="#">ACQUIRE:CURRENTCOUNT:HISTWFMS?</a>	Returns histogram waveforms count
<a href="#">ACQUIRE:CURRENTCOUNT:MASKHITS&lt;x&gt;?</a>	Returns mask hits count
<a href="#">ACQUIRE:CURRENTCOUNT:MASKSAMPLES?</a>	Returns mask samples count
<a href="#">ACQUIRE:CURRENTCOUNT:MASKTHITS?</a>	Returns total mask hits count
<a href="#">ACQUIRE:CURRENTCOUNT:MASKWFMS?</a>	Returns mask waveforms count
<a href="#">ACQUIRE:DATA:CLEAR</a>	Clears all acquired data
<a href="#">ACQUIRE:MODE</a>	Sets or returns acquisition mode
<a href="#">ACQUIRE:NUMAVG</a>	Sets or returns number of acquisitions for an averaged waveform
<a href="#">ACQUIRE:SAVEFILE:SAVESCREEN</a>	Sets or returns the file to save screen to on acquisition stopped

Table 2-17: Acquisition Commands (cont.)

Command	Description
ACQUIRE:SAVEFile:SAVEWfm	Sets or returns the file to save waveform data to on acquisition stopped
ACQUIRE:STATE	Starts, stops, or returns acquisition state
ACQUIRE:STOPAfter?	Returns all stopafter parameters
ACQUIRE:STOPAfter:ACTion	Sets or returns the stopafter action
ACQUIRE:STOPAfter:BELL	Sets or returns whether to sound bell on acquisition stopped
ACQUIRE:STOPAfter:CONDition	Sets or returns the acquisition stopafter condition
ACQUIRE:STOPAfter:COUNt	Sets or returns the stopafter count value The stopafter count value is dependent on the condition set by the ACQUIRE:STOPAfter:CONDition command
ACQUIRE:STOPAfter:MODE	Sets or returns the stopafter mode

## Calibration Command Group

The calibration commands provide information about the current state of the calibration for the mainframe and all resident sampling-module channels. Additional commands allow you to update portions of the “electronic calibration sticker” information, to check the protection status of the calibration information, and to set or query the front-panel DC calibration output.

Command	Description
CALIBRATE:DATE:CH<x>?	Returns date and time of the last sampling module channel<x> calibration
CALIBRATE:DATE:MAInframe?	Returns the date and time of the last mainframe calibration
CALIBRATE:DCCALibrator	Sets or returns the value of the DC Calibrator voltage
CALIBRATE:HOSTInfo:CH<x>?	Returns the mainframe model number, serial number, and mainframe channel<x> in which the sampling module channel was located during the last calibration information update.
CALIBRATE:LOCK:STATus?	Returns the status of the calibration protection mode.
CALIBRATE:STATus:CH<x>?	Returns calibration status for specified sampling module channel<x>
CALIBRATE:STATus:MAInframe?	Returns calibration status for mainframe



Command	Description
<a href="#">CALibrate:TEMPerature:CH&lt;x&gt;?</a>	Returns the difference in °C between the current sampling module channel<x> temperature and the temperature recorded at the last calibration information update
<a href="#">CALibrate:TEMPerature:MAInframe?</a>	Returns the difference in °C between the current mainframe temperature and the temperature recorded at the last calibration information update
<a href="#">CALibrate:UPDATEinfo:ALLModules</a>	Updates date, time, temp and host mainframe information for all sampling modules
<a href="#">CALibrate:UPDATEinfo:CH&lt;x&gt;</a>	Updates the calibration information in the nonvolatile memory of the sampling module channel<x>
<a href="#">CALibrate:UPDATEinfo:MAInframe</a>	Updates the calibration information in the nonvolatile memory of the mainframe

## Compensation Command Group

The compensation commands provide information about the current state of the compensation for the mainframe and all installed module channels, means to invoke compensation functions, and management of compensation storage memory locations.

There are two nonvolatile compensation storage memory blocks in the mainframe and each sampling module channel: Factory and User. In addition, there is a volatile run-time, in-use version of all compensation data, which is the compensation data actually used during the operation of the instrument. On power-up the instrument loads the User compensation data into the runtime compensation array.

Command	Description
<a href="#">COMPensate:ALLModules</a>	Compensates all installed modules
<a href="#">COMPensate:CH&lt;x&gt;</a>	Compensates the module channel<x> for DC variances
<a href="#">COMPensate:DARKLev:CH&lt;x&gt;</a>	Compensates the specified optical channel<x> by removing residual DC offsets in the entire vertical path
<a href="#">COMPensate:DATE:CH&lt;x&gt;?</a>	Returns date and time of the current in-use compensation data for the module channel<x>
<a href="#">COMPensate:DATE:MAInframe?</a>	Returns date and time of the current in-use compensation data for the mainframe
<a href="#">COMPensate:MAInframe</a>	Compensates the mainframe for DC variances

Command	Description
<a href="#">COMPensate:OPTGAIN:CH&lt;x&gt;</a>	Compensates optical user wavelength gain (wavelengths and input power)
<a href="#">COMPensate:RECAI:FACTory:ALLModules</a>	Recalls compensation data for all installed module channels from their nonvolatile factory memories into run-time compensation memory
<a href="#">COMPensate:RECAI:FACTory:CH&lt;x&gt;</a>	Recalls compensation data from the nonvolatile factory memory of the module channel<x> into its associated run-time memory
<a href="#">COMPensate:RECAI:FACTory:MAInframe</a>	Recalls compensation data from the nonvolatile factory memory in the mainframe into its associated run-time memory
<a href="#">COMPensate:RECAI:USER:ALLModules</a>	Recalls compensation data for all installed module channels from their respective nonvolatile user memories into run-time compensation memory
<a href="#">COMPensate:RECAI:USER:CH&lt;x&gt;</a>	Recalls compensation data from the nonvolatile user memory of the module channel<x> into its associated run-time memory
<a href="#">COMPensate:RECAI:USER:MAInframe</a>	Recalls compensation data from the nonvolatile user memory in the mainframe into its associated run-time memory
<a href="#">COMPensate:RESults?</a>	Returns brief explanation of the results of the last compensation
<a href="#">COMPensate:RESults:VERBoSe?</a>	Returns results of last compensation, with a more detailed explanation than the COMPensate:RESults? query returns
<a href="#">COMPensate:SAVe:USER:ALLModules</a>	Saves volatile run-time compensation data for all installed modules into their associated nonvolatile user memories
<a href="#">COMPensate:SAVe:USER:CH&lt;x&gt;</a>	Saves volatile run-time compensation data for the module channel<x> into its nonvolatile user memory
<a href="#">COMPensate:SAVe:USER:MAInframe</a>	Saves volatile run-time compensation data for the mainframe into its nonvolatile user memory
<a href="#">COMPensate:STATus:CH&lt;x&gt;?</a>	Returns the current compensation status of the specified module channel<x>
<a href="#">COMPensate:STATus:MAInframe?</a>	Returns the current compensation status of the mainframe

Command	Description
<a href="#">COMPensate:TEMPerature:CH&lt;x&gt;?</a>	Returns the difference in °C between the current temperature of the module channel<x> and the temperature currently residing in its in-use run-time compensation memory
<a href="#">COMPensate:TEMPerature:MAInframe?</a>	Returns the difference in °C between the current mainframe temperature and the temperature currently residing in its in-use run-time compensation memory

## Cursor Command Group

Use the commands in the Cursor Command Group to control the cursor display and readout. You can use these commands to control the setups for cursor 1 and cursor 2, such as waveform source, cursor position, and cursor color.

You can also use the commands to select one of the following cursor functions:

- **Off** Shuts off the display of all cursors.
- **Vertical Bars.** Displays vertical bar cursors, which provide traditional horizontal unit readouts for Cursor 1 (bar1), Cursor 2 (bar2), the delta between them, and 1/delta (results in frequency when the horizontal unit is time).
- **Horizontal Bars.** Displays horizontal bar cursors, which provide traditional vertical unit readouts for Cursor 1 (bar1), Cursor 2 (bar2), and the delta between them.
- **Waveform.** Displays waveform cursors, which provide horizontal and vertical unit readouts for Cursor 1 (bar1), Cursor 2 (bar2), the delta between them, and 1/delta (results in frequency when the horizontal unit is time).

Command	Description
<a href="#">CURSor?</a>	Returns all cursor parameters
<a href="#">CURSor:CURSor&lt;x&gt;:COLOR</a>	Sets or returns cursor<x> color
<a href="#">CURSor:CURSor&lt;x&gt;:SOURce</a>	Sets or returns cursor<x> waveform source and timebase
<a href="#">CURSor:FUNction</a>	Sets or returns the cursor type
<a href="#">CURSor:HBArs?</a>	Returns hbar cursor parameters
<a href="#">CURSor:HBArs:DELTA?</a>	Returns hbars cursors vertical difference
<a href="#">CURSor:HBArs:POSition&lt;x&gt;</a>	Sets or returns the hbar cursor<x> vertical position
<a href="#">CURSor:SElect</a>	Sets or returns which cursor is active for front-panel control
<a href="#">CURSor:VBArs?</a>	Returns vbar cursor parameters
<a href="#">CURSor:VBArs:DELTA?</a>	Returns the difference between vbar cursors

Command	Description
<a href="#">CURSor:VBArS:POSition&lt;x&gt;</a>	Sets or returns the vbar cursor<x> horizontal position
<a href="#">CURSor:WAVeform?</a>	Returns waveform cursor parameters
<a href="#">CURSor:WAVeform:HDELTA?</a>	Returns the horizontal difference between waveform cursors
<a href="#">CURSor:WAVeform:HPOS&lt;x&gt;?</a>	Returns the position of waveform cursor <x>
<a href="#">CURSor:WAVeform:POSition&lt;x&gt;</a>	Sets or returns the position of waveform cursor <x>
<a href="#">CURSor:WAVeform:VDELTA?</a>	Returns the vertical difference between waveform cursors

## Display Control Command Group

You use the commands in the Display Control Command Group to change the graticule style, the displayed intensities, and to set the characteristics of the waveform display.

You can set the following:

- Background color (default is black) and foreground color (default is silver).
- Cursor, histogram, mask, and measurement annotation colors.
- Whether cursor, histogram, mask, and measurement readouts are displayed.
- Whether measurement annotations are displayed.
- Whether waveforms are simply displayed in Normal mode as dots or vectors, in Variable Persistence mode, or in Infinite Persistence mode.
- Whether the instrument uses interpolation to increase sample density of waveform for record lengths less than 1000 points, and, if interpolation is used, which type (Sin(x) or Linear).
- The style of graticule that underlies the waveforms.

Use the commands to set the style that best displays your waveforms and graticule display properties. The mode you choose globally affects all displayed waveforms; for example, you cannot set channel 1 to display in Normal mode and channel 2 in Variable Persistence mode.

There are four graticule settings:

- Frame
- Grid
- Cross Hair
- Full

Choose Frame or Grid for minimum clutter on screen; choose Full or Cross Hair for ease in taking graticule measurements.

Command	Description
DISplay?	Returns current display settings
DISplay:COLor?	Returns color group settings
DISplay:COLor:BACKground	Sets or returns graticule background color
DISplay:COLor:CURSor<x>	Sets or returns cursor<x> color
DISplay:COLor:FOREground	Sets or returns graticule foreground color
DISplay:COLor:HIStogram	Sets or returns histogram rectangle and plot color
DISplay:COLor:MASK	Sets or returns the color of mask polygons
DISplay:CURSReadout	Sets or returns the display state of the cursor readout
DISplay:DATE	Turns the Date/Time display on or off or returns the status of the Date/Time display
DISplay:GRAticule?	Returns all graticule parameters
DISplay:GRAticule:HDIVS?	Returns the number of horizontal divisions in graticule
DISplay:GRAticule:STYLE	Sets or returns the graticule style
DISplay:GRAticule:VDIVS?	Returns the number of vertical divisions in graticule
DISplay:HISTRReadout	Sets or returns the display state of the histogram readout
DISplay:INTERPolat	Sets or returns the display interpolation type
DISplay:MASKReadout	Sets or returns the display state of the mask readout
DISplay:MEASBar	Sets or returns the display state of the measurement bar
DISplay:MEASReadout	Sets or returns the display state of the measurement readout
DISplay:PERsistence	Sets or returns the display persistence time
DISplay:SHOWVector	Sets or returns the show vector status
DISplay:STYle	Sets or returns the display persistence style
DISplay:WFMReadout	Sets or returns the display state of the waveform readout

## Hard Copy Command Group

Hard copy commands allow you to make hard copies of your data file or send hard copy data in various formats to a specified file.

Command	Description
<a href="#">HARDCopy</a>	Sends a screen copy to the selected port or returns the selected port and file path
<a href="#">HARDCopy:FILENAME</a>	Sets or returns the hard copy file path
<a href="#">HARDCopy:FORMAT</a>	Selects the file format when sending a hardcopy to a file using the <a href="#">HARDCopy:FILENAME</a> command
<a href="#">HARDCopy:INKSaver</a>	Sets the Ink-saver mode on or off. Ink-saver mode can conserve ink and improve print quality when printing images of waveform displays

## Histogram Command Group

Histogram commands let you select the type of histogram, what part of the waveform should go into the histogram, and histogram statistics. You can use commands from this group to do the following:

- Select any channel, math, or reference waveform and create a histogram of vertical or horizontal values for it.
- Adjust the limits of the box that define the area on the waveform from which the histogram data is obtained. The histogram box can be set using source waveform coordinates or percentage-of-display coordinates.
- Create a linear or logarithmic plot of histogram data and set plot size and color.
- Clear histogram count and restart.
- Turn the display of the histogram on or off.
- Set or query the color of the histogram box and histogram plot.
- Enable or disable histogram calculations.
- Get histogram statistics, such as total hits, mean value, peak-to-peak value, and standard deviation.
- Get all the histogram parameters.

---

**NOTE.** You can also export a histogram to a file of comma-separated values. See the [EXPort](#) command for more information.

---

Command	Description
<a href="#">HISTogram?</a>	Return all histogram parameters
<a href="#">HISTogram:BOX</a>	Sets or returns the left, top, right, and bottom positions of the histogram box, in source waveform coordinates

Command	Description
<a href="#">HIStogram:BOXPcnt</a>	Sets or returns same as HIStogram:BOX, but in percentage coordinates, with 0,0 upper left and 100,100 lower right
<a href="#">HIStogram:COLOr</a>	Sets or returns the histogram color
<a href="#">HIStogram:COUnT</a>	Clears histogram count source data and restarts counting
<a href="#">HIStogram:DISPlay</a>	Sets or returns whether histogram data is displayed on screen
<a href="#">HIStogram:ENABle</a>	Enables or disables histogram calculations Returns whether histogram calculations are enabled
<a href="#">HIStogram:MODe</a>	Sets type of histogram to be done, either vertical or horizontal Returns the type of histogram
<a href="#">HIStogram:SIze</a>	Sets or returns the width (or height) of the histogram on the screen in divisions
<a href="#">HIStogram:SOUrcE</a>	Sets or returns the source waveform and timebase (Main, Mag1, or Mag2) for histogram
<a href="#">HIStogram:STATistics?</a>	Returns all histogram statistics
<a href="#">HIStogram:STATistics:HITS?</a>	Returns the histogram total hits value
<a href="#">HIStogram:STATistics:MEAN?</a>	Returns the histogram mean value
<a href="#">HIStogram:STATistics:MEDIAN?</a>	Returns the histogram median value
<a href="#">HIStogram:STATistics:PEAKHits?</a>	Returns the histogram peak hits value
<a href="#">HIStogram:STATistics:PKTOPK?</a>	Returns the histogram peak to peak value
<a href="#">HIStogram:STATistics:SIGMA&lt;x&gt;?</a>	Returns population density for $\pm<x>$ sigma value
<a href="#">HIStogram:STATistics:STDdev?</a>	Returns the histogram standard deviation value
<a href="#">HIStogram:STATistics:WAVEforms?</a>	Returns the number of waveforms used in histogram
<a href="#">HIStogram:TYPE</a>	Sets or returns whether the histogram is displayed linearly or logarithmically
<a href="#">HIStogram:WFMDB:STATE</a>	Sets or returns whether the histogram counting is on a waveform database

## Horizontal Command Group

You use the commands from the Horizontal Command Group to control the timebases of the instrument. You can use these commands to do the following:

- Set the scale (time, distance or bits per division or screen) of the Main, Mag1, and Mag2 timebases.
- Set the record lengths for the Main, Mag1, and Mag2 timebases.
- Get the time of first point and time of last point for the Main, Mag1, and Mag2 timebases.
- Get the sample resolution of the Main, Mag1, and Mag2 timebases.
- Set the horizontal position for the Main, Mag1, and Mag2 timebases.
- Set the horizontal reference for the Main, Mag1, and Mag2 timebases.
- Enable or disable the acquisition and display of the Mag1 and Mag2 timebases.
- Set timebase units to seconds, bits, or distance.
- Set the Dielectric constant or propagation velocity (value is used to convert time to distance when distance is specified for horizontal units).
- Select a communication standard, such as OC192, that automatically sets the associated bit rate.
- Adjust the external 10 MHz reference frequency to ensure that the TDR timebase locks to the signal.
- Set the parameters for FrameScan mode, and turn the mode on or off.
- Get the screen resolution of the Main, Mag1, and Mag2 timebases.
- Get all the horizontal settings.

Command	Description
<a href="#">AUTOSet:HORizontal</a>	Sets or returns the status for the horizontal Autoset options
<a href="#">HORizontal?</a>	Returns all horizontal settings
<a href="#">HORizontal:BITS:BITRate</a>	Sets or returns the bit rate of the timebase
<a href="#">HORizontal:BITS:STANdard</a>	Sets or returns the communication standard (or NONE) for the bit rate
<a href="#">HORizontal:DISTance:DIElectric</a>	Sets or returns the dielectric constant
<a href="#">HORizontal:DISPlayscale:BITS</a>	sets or queries the display scale mode when horizontal units are set to BITS.
<a href="#">HORizontal:DISPlayscale:DISTance</a>	sets or queries the display scale mode when horizontal units are set to distance (meters, feet or inches).
<a href="#">HORizontal:DISPlayscale:SEConds</a>	Sets or queries the display scale mode when horizontal units are set to SEConds.
<a href="#">HORizontal:DISTance:PVELOCITY</a>	Sets or returns the propagation velocity
<a href="#">HORizontal:EXT10MHZref:FREQ</a>	Sets or returns the frequency of the 10 MHz external reference when enabled for TDR acquisitions



Command	Description
<a href="#">HORizontal:FRAMescan:RESET</a>	Resets FrameScan acquisition
<a href="#">HORizontal:FRAMescan:SCANBits</a>	Sets or returns the number of bits in frame to scan in FrameScan acquisition mode
<a href="#">HORizontal:FRAMescan:STATE</a>	Sets or returns the FrameScan acquisition mode on or off
<a href="#">HORizontal:MAGnify&lt;x&gt;?</a>	Returns all Mag<x> timebase settings
<a href="#">HORizontal:MAGnify&lt;x&gt;:POSition</a>	Sets or returns the horizontal position for Mag<x> timebase
<a href="#">HORizontal:MAGnify&lt;x&gt;:RECOrdlength</a>	Sets or returns the Mag<x> timebase record length
<a href="#">HORizontal:MAGnify&lt;x&gt;:REFPoint</a>	Sets or returns the Mag<x> timebase reference point in percent
<a href="#">HORizontal:MAGnify&lt;x&gt;:RESolution?</a>	Returns the Mag<x> timebase acquisition resolution
<a href="#">HORizontal:MAGnify&lt;x&gt;:SCAle</a>	Sets or returns the Mag<x> timebase time per division
<a href="#">HORizontal:MAGnify&lt;x&gt;:TOFPoint?</a>	Returns the Mag<x> timebase time of first point
<a href="#">HORizontal:MAGnify&lt;x&gt;:TOLPoint?</a>	Returns the Mag<x> timebase time of last point
<a href="#">HORizontal:MAGnify&lt;x&gt;:VIEW</a>	Sets or returns the Mag<x> timebase view on or off
<a href="#">HORizontal:MAIn?</a>	Returns the time per division of the main time base
<a href="#">HORizontal:MAIn:BITS:POSition</a>	Sets or queries the horizontal position for the main timebase in bits.
<a href="#">HORizontal:MAIn:BITS:SCAle</a>	Sets or queries the scale (bits per division) for the Main timebase.
<a href="#">HORizontal:MAIn:POSition</a>	Sets or returns the horizontal position for the main timebase
<a href="#">HORizontal:MAIn:RECOrdlength</a>	Sets or returns the main timebase record length
<a href="#">HORizontal:MAIn:REFPoint</a>	Sets or returns the main timebase reference position in percent of record
<a href="#">HORizontal:MAIn:RESolution?</a>	Returns the main timebase acquisition resolution
<a href="#">HORizontal:MAIn:SCAle</a>	Sets or returns the main timebase time per division
<a href="#">HORizontal:MAIn:TOFPoint?</a>	Returns the main timebase time of first point
<a href="#">HORizontal:MAIn:TOLPoint?</a>	Returns the main timebase time of last point
<a href="#">HORizontal:MATH&lt;x&gt;:MAGnify&lt;x&gt;:POSition?</a>	Returns the Math<x>"Acquisition" horizontal position for Mag<x> timebase

Command	Description
<a href="#">HORizontal:MATH&lt;x&gt;:MAGnify&lt;x&gt;:RECOrdlength?</a>	Returns the Math<x> Mag<x> timebase record length
<a href="#">HORizontal:MATH&lt;x&gt;:MAGnify&lt;x&gt;:RESolution?</a>	Returns the Math<x> Mag<x> timebase acquisition resolution
<a href="#">HORizontal:MATH&lt;x&gt;:MAGnify&lt;x&gt;:SCALE?</a>	Returns the Math<x> Mag<x> timebase time per division
<a href="#">HORizontal:MATH&lt;x&gt;:MAGnify&lt;x&gt;:TOFPoInt?</a>	Returns the Math<x> Mag<x> timebase time of first point
<a href="#">HORizontal:MATH&lt;x&gt;:MAGnify&lt;x&gt;:TOLPoInt?</a>	Returns the Math<x> Mag<x> timebase time of last point
<a href="#">HORizontal:MATH&lt;x&gt;:MAIn:POSition?</a>	Returns the Math<x> horizontal position for main timebase
<a href="#">HORizontal:MATH&lt;x&gt;:MAIn:RECOrdlength?</a>	Returns the Math<x> main timebase record length
<a href="#">HORizontal:MATH&lt;x&gt;:MAIn:REFPoInt?</a>	Returns the Math<x> main timebase reference position in percent of record
<a href="#">HORizontal:MATH&lt;x&gt;:MAIn:RESolution?</a>	Returns the Math<x> main timebase acquisition resolution
<a href="#">HORizontal:MATH&lt;x&gt;:MAIn:SCALE?</a>	Returns the Math<x> main timebase time per division
<a href="#">HORizontal:MATH&lt;x&gt;:MAIn:TOFPoInt?</a>	Returns the Math<x> main timebase time of first point
<a href="#">HORizontal:MATH&lt;x&gt;:MAIn:TOLPoInt?</a>	Returns the Math<x> main timebase time of last point
<a href="#">HORizontal:REF&lt;x&gt;:MAGnify&lt;x&gt;:POSition?</a>	Returns the Reference<x> "Acquisition" horizontal position for Mag<x> timebase
<a href="#">HORizontal:REF&lt;x&gt;:MAGnify&lt;x&gt;:RECOrdlength?</a>	Returns the Reference<x> Mag<x> timebase record length
<a href="#">HORizontal:REF&lt;x&gt;:MAGnify&lt;x&gt;:RESolution?</a>	Returns the Reference<x> Mag<x> timebase screen resolution
<a href="#">HORizontal:REF&lt;x&gt;:MAGnify&lt;x&gt;:SCALE?</a>	Returns the Reference<x> Mag<x> timebase time per division
<a href="#">HORizontal:REF&lt;x&gt;:MAGnify&lt;x&gt;:TOFPoInt?</a>	Returns the Reference<x> Mag<x> timebase time of first point
<a href="#">HORizontal:REF&lt;x&gt;:MAGnify&lt;x&gt;:TOLPoInt?</a>	Returns the Reference<x> Mag<x> timebase time of last point
<a href="#">HORizontal:REF&lt;x&gt;:MAIn:POSition?</a>	Returns the Reference<x> "Acquisition" horizontal position for main timebase
<a href="#">HORizontal:REF&lt;x&gt;:MAIn:RECOrdlength?</a>	Returns the Reference<x> main timebase record length
<a href="#">HORizontal:REF&lt;x&gt;:MAIn:REFPoInt?</a>	Returns the Reference<x> main timebase reference position in percent of record

Command	Description
<a href="#">HORizontal:REF&lt;x&gt;:MAIn:RESolution?</a>	Returns the Reference<x> main timebase screen resolution
<a href="#">HORizontal:REF&lt;x&gt;:MAIn:SCAlle?</a>	Returns the Reference<x> main timebase time per division
<a href="#">HORizontal:REF&lt;x&gt;:MAIn:TOFPoInt?</a>	Returns the Reference<x> main timebase time of first point
<a href="#">HORizontal:REF&lt;x&gt;:MAIn:TOLPoInt?</a>	Returns the Reference<x> main timebase time of last point
<a href="#">HORizontal:UNIts</a>	Sets or returns the horizontal units

## Mask Command Group

Mask commands control standard masks, user-defined masks, and testing against masks. A mask is a set of polygonal regions on the screen. Every vertical line on the screen intersects the polygon in zero, one, or two places, but never in more than two places. (A vertical line that intersects a vertical mask border is counted.) You have to break up more complicated polygons into two separate masks. Unlike limit testing, the inside of a mask is the region where waveform data would not normally fall.

A telecommunications standard may require up to eight of these masks. Pulse standards always have two masks. Standards with eye patterns usually have three masks, but some have four.

You use the commands in the Mask Command Group to do the following:

- Specify the waveform source to test and the mask to use.
- Specify whether to use, and the size of, mask margins, which allow you to shrink or expand an existing set of polygons by a specified percentage.
- Specify whether to display a readout of hits and the mask on screen. Options also exist for autsetting the incoming waveforms to match the mask you choose.
- Select industry-standard masks that support a variety of electrical and optical communication standards.
- Define and edit your own custom mask; create an entirely new mask, or use a standard mask as a starting reference, and edit it to meet your needs.
- Enable, disable, or reset the mask counts. Once you turn on mask counting, it remains on until you explicitly turn it off.
- Set the color for the mask polygons.

Command	Description
<a href="#">MASK?</a>	Return all mask parameters
<a href="#">MASK:AUTOSet:MODE</a>	Sets or returns the mask autoselect mode. If AUTO is set, a mask autoselect will automatically be done after a standard mask is selected; if MANual is set, mask autoselect for a standard mask runs only if the user presses the AUTOSET button or sends the AUTOSet EXECute command.
<a href="#">MASK:AUTOSet:HILOW:METHOD</a>	Sets or returns the method, Mean or Mode, that a Mask Autoselect uses to determine the High and Low values
<a href="#">MASK:COLOR</a>	Sets or returns the mask color
<a href="#">MASK:COUNT</a>	Clear mask counts and source data, and restart counting Returns all the values for the mask count parameters
<a href="#">MASK:COUNT:SAMPLES?</a>	Returns the total number of sample points that have gone into mask counting
<a href="#">MASK:COUNT:STATE</a>	Sets or returns the mask counting
<a href="#">MASK:COUNT:TOTAL?</a>	Returns the sum of all hits in all mask polygons
<a href="#">MASK:COUNT:WAVEFORMS?</a>	Returns the number of waveforms used in the mask
<a href="#">MASK:DISPLAY</a>	Sets or returns whether or not defined masks are displayed on the screen
<a href="#">MASK:MARGIN:PERCENT</a>	Sets or returns the mask margin in percent
<a href="#">MASK:MARGIN:STATE</a>	Sets or returns the mask margins state
<a href="#">MASK:MASK&lt;x&gt;</a>	Delete all points in mask<x> Returns all mask<x>parameters
<a href="#">MASK:MASK&lt;x&gt;:COUNT?</a>	Returns number of hits in mask<x>
<a href="#">MASK:MASK&lt;x&gt;:NR_PT?</a>	Returns number of points in mask<x>
<a href="#">MASK:MASK&lt;x&gt;:POINTS</a>	Returns the points in the specified mask in waveform coordinates
<a href="#">MASK:MASK&lt;x&gt;:POINTSPCNT</a>	Sets or returns the points in mask<x>, in percentage coordinates, with 0,0 upper left and 100,100 lower right
<a href="#">MASK:SOURCE</a>	Sets or returns which waveform and timebase will be compared against the mask(s) when counting is turned on
<a href="#">MASK:STANDARD</a>	Sets or returns the standard communication mask
<a href="#">MASK:WFMDB:STATE</a>	Returns whether a waveform database is used as a source for mask counting

## Math Command Group

You use the commands in the Math Command Group to create and define math waveforms. You can define and display up to eight math waveforms simultaneously. You use the available math functions, such as integration, differentiation, square root, and natural logs, to define your math waveform.

Math expressions can be simple, such as C1, which specifies that a waveform should show the signal source of channel 1 with no mathematical computation. Math expressions can also be complex, consisting of 100 plus characters and comprising many sources, functions, and operators.

Math expressions require at least one source waveform. When the acquisition of a live waveform stops, so does the acquisition of any math waveforms using that waveform as a source. When a live waveform update occurs or reference waveform is altered, math waveforms containing those waveforms as sources are also updated to reflect the changes. Also, sources must exist, but do not need to be displayed to be used in and to update math waveforms.

Command	Description
<a href="#">MATH&lt;x&gt;?</a>	Returns math<x> settings
<a href="#">MATH&lt;x&gt;:DEFine</a>	Sets or returns the math<x> definition
<a href="#">MATH&lt;x&gt;:FILTer:MODE</a>	Sets or returns the filter mode for the math<x> waveform
<a href="#">MATH&lt;x&gt;:FILTer:RISetime</a>	Sets or returns the risetime (bandwidth) of the math filter function
<a href="#">MATH&lt;x&gt;:NUMavg</a>	Sets or returns the number of waveforms to average for a math waveform for the math<x> waveform
<a href="#">MATH&lt;x&gt;:POSition</a>	Sets or returns the math<x> vertical position
<a href="#">MATH&lt;x&gt;:SCAlE</a>	Sets or returns the math<x> vertical scale (per div)
<a href="#">MATH&lt;x&gt;:UNIts?</a>	Returns math units
<a href="#">MATH&lt;x&gt;:WFMLabel</a>	Sets or returns the label associated with the math<x> waveform

## Measurement Command Group

You use the commands in the Measurement Command Group to control the automated measurement system. Up to eight automated measurements can be displayed on the screen. In the commands, these eight measurement slots are named MEAS<x>, where <x> can be 1 through 8. You use the commands to do the following:

- Obtain measurement results.
- Set and query measurement parameters. You can assign most parameters differently for each source of a measurement slot.
- Select the measurement slot (1 through 8), and turn it on and off.
- Select the waveform (Source1) to be measured (or the Source1 and Source2 waveforms for delay and other two-waveform measurements).
- Query the value of a specified measurement.
- Clear the selected measurement and its statistics.
- Select whether the measurement displays annotations (indicating which portion of the waveform is being measured as well as reference levels for that measurement) and statistics.
- Select whether or not statistics on measurements are computed.
- Perform measurements on waveform databases.
- Set the signal type for waveform database measurements (Pulse, Eye, or RZ).
- Clear the waveform database.
- Define measurement regions using gates.
- Set slope and direction for delay measurements.
- Select a tracking method (algorithm) that is used to track the high and low value of the waveform.
- Enable tracking of the high and low values of the waveform automatically, and specify a high and/or low value (when tracking is disabled).
- Select a reference level calculation method.
- Set Hi, Mid, and Low reference values, either as percentages of the high-low range or as absolute values.
- Set measurement parameters to default values.

Command	Description
<a href="#">MEASUrement?</a>	Returns all measurement parameters
<a href="#">MEASUrement:ALL:VALue?</a>	Returns all measurement values
<a href="#">MEASUrement:ANNOtations:STATE</a>	Sets or returns whether the measurement shows annotations
<a href="#">MEASUrement:LIST</a>	Sets or returns a list of defined measurements for which you want values returned
<a href="#">MEASUrement:LISTValue?</a>	Returns the values of the measurements in the list created with the MEASUrement:LIST command

Command	Description
MEASUrement:MEAS<x>:ALL?	Returns all measurement statistics values for the measurement specified by x
MEASUrement:MEAS<x>:EYEWindow	Sets or returns the percent of interval between two eye crossings centered on the middle of the region
MEASUrement:MEAS<x>:GATing:STATE	Sets or returns the gating state (on or off) for the measurement specified by x
MEASUrement:MEAS<x>:JITter	Sets or returns the jitter-level crossing for measurement
MEASUrement:MEAS<x>:MAXimum?	Returns measurement statistics maximum value for measurement <x>
MEASUrement:MEAS<x>:MEAN?	Returns measurement statistics mean value for measurement <x>
MEASUrement:MEAS<x>:MINimum?	Returns measurement statistics minimum value for measurement <x>
MEASUrement:MEAS<x>:NOISe	Sets or returns whether noise is measured on the high or low level of the signal
MEASUrement:MEAS<x>:REFLevel<x>?	Returns all reference level <x> settings for measurement slot <x>
MEASUrement:MEAS<x>:REFLevel<x>:ABSolute:HIGH	Sets or returns the top reference level in absolute waveform units
MEASUrement:MEAS<x>:REFLevel<x>:ABSolute:LOW	Sets or returns the low reference level in absolute waveform units
MEASUrement:MEAS<x>:REFLevel<x>:ABSolute:MID	Sets or returns the mid reference level in absolute waveform units
MEASUrement:MEAS<x>:REFLevel<x>:METHod	Sets or returns the method to calculate reference levels, either as a % of the high-low range or in absolute vertical units
MEASUrement:MEAS<x>:REFLevel<x>:RELative:HIGH	Sets or returns the high reference level as a % of the high-low range
MEASUrement:MEAS<x>:REFLevel<x>:RELative:LOW	Sets or returns the low reference level as a % of the high-low range
MEASUrement:MEAS<x>:REFLevel<x>:RELative:MID	Sets or returns the mid reference level as a % of the high-low range
MEASUrement:MEAS<x>:SETDefault	Sets all measurement values to the instrument default settings
MEASUrement:MEAS<x>:SOUrce<x>:EDGE?	:EDGE? Returns all edge settings for the specified measurement
MEASUrement:MEAS<x>:SOUrce<x>:EDGE:DIRection	Sets or returns the direction (forward or backward) that the instrument uses to look for the rising or falling edge
MEASUrement:MEAS<x>:SOUrce<x>:EDGE:SLOPe	Sets or returns the slope of the edges used in delay time measurements

Command	Description
MEASUREMENT:MEAS<x>:SOURCE<x>:GATE<x>?	Returns the specified gate<x> settings for measurement<x>
MEASUREMENT:MEAS<x>:SOURCE<x>:GATE<x>:PCTPOS	Sets or returns the gate endpoint in percent
MEASUREMENT:MEAS<x>:SOURCE<x>:GATE<x>:POS	Sets or returns the gate endpoint in waveform units
MEASUREMENT:MEAS<x>:SOURCE<x>:HILow?	Returns all high / low values for measurement<x>
MEASUREMENT:MEAS<x>:SOURCE<x>:HILow:METHod	Sets or returns the method for calculating high / low levels
MEASUREMENT:MEAS<x>:SOURCE<x>:HILow:TRACk:HIGH:ENABle	Sets or returns the tracking high level (on or off)
MEASUREMENT:MEAS<x>:SOURCE<x>:HILow:TRACk:HIGH:VALue	Sets or returns the high value used to calculate a specified measurement on a specified source waveform
MEASUREMENT:MEAS<x>:SOURCE<x>:HILow:TRACk:LOW:ENABle	Sets or returns the tracking low level (on or off)
MEASUREMENT:MEAS<x>:SOURCE<x>:HILow:TRACk:LOW:VALue	Sets or returns the low value used to calculate a specified measurement on a specified source waveform
MEASUREMENT:MEAS<x>:SOURCE<x>:WFM	Sets or returns the measurement source waveform, and, optionally, the timebase on which measurements are taken
MEASUREMENT:MEAS<x>:SOURCE<x>:WFMDB:SIGType	Sets or returns the signal type of the waveform database for the measurement source
MEASUREMENT:MEAS<x>:SOURCE<x>:WFMDB:STATE	Sets or returns the state of Use WfmDb for measurement source (on or off)
MEASUREMENT:MEAS<x>:STATE	Sets or returns the display of measurement (on or off)
MEASUREMENT:MEAS<x>:STATIStics:CLEAr	Clears measurement statistics for measurement<x>
MEASUREMENT:MEAS<x>:STDdev?	Returns measurement statistics standard deviation value for measurement<x>
MEASUREMENT:MEAS<x>:TYPE	Sets or returns the type of measurement for the specified measurement
MEASUREMENT:MEAS<x>:UNIts?	Returns the units for the specified measurement
MEASUREMENT:MEAS<x>:VALue?	Returns the measurement value for the specified measurement
MEASUREMENT:STATIStics:ENABle	Sets or returns whether or not measurement statistics are enabled (on or off)
MEASUREMENT:STATIStics:WEIghing	Sets or returns measurement statistics weighting for all measurements



## Miscellaneous Command Group

Miscellaneous commands do not fit into other categories. Several commands and queries are common to all 488.2–1987 devices on the GPIB bus. The 488.2–1987 standard defines these commands. The common commands begin with an asterisk (\*) character.

Command	Description
<a href="#">APPLication:ACTivate</a>	Launches the available optional software applications
<a href="#">AUTOSet</a>	Runs autosest
<a href="#">AUTOSet:STOP</a>	Stops autosest
<a href="#">AUTOSet:TYPE</a>	Sets or returns the autosest mode (Edge, Period, NRZ Eye, RZ Eye, or TDR)
<a href="#">AUTOSet:UNDO</a>	Undoes autosest
<a href="#">DATE</a>	Sets or returns the date (yyyy-mm-dd)
<a href="#">FACTory</a>	Sets instrument to factory defaults
<a href="#">FILESystem:READFile?</a>	Outputs the specified file to the GPIB port with maximum file size of 10 Mb
<a href="#">HEADer</a>	Sets or returns the Response Header Enable State
<a href="#">*IDN?</a>	Returns identification string
<a href="#">LOCK</a>	Sets or returns the front panel lock state
<a href="#">*LRN?</a>	Returns "complete" instrument settings
<a href="#">*PSC</a>	Sets the power-on clear flag. When false, registers retain their status when power is restored; when true, registers will be cleared when power is restored
<a href="#">SET?</a>	Tek Learn Mode
<a href="#">SYNC:TIMEOUT</a>	Sets or returns the default synchronization timeout for commands that rely upon data production from the instrument
<a href="#">TIMe</a>	Sets or returns the time displayed by the instrument
<a href="#">UNLock</a>	Unlocks the front panel Returns the front panel lock state
<a href="#">VERBose</a>	Sets or returns the response header abbreviation control command

## Phase Reference Command Group

You use the commands in the Phase Reference Command Group to switch to the Phase Correction timebase. This timebase supports ultra-low trigger jitter (typical is 200 fsec RMS), improving the fidelity of acquired signals.

The 82A04 module is available in two configurations: standard and with Option 60G. You can find 82A04 information by clicking the Help button in the Phase Ref Setup dialog box

Command	Description
<a href="#">PHAseref:CHAR</a>	Initiates a Phase Reference Characterization
<a href="#">PHAseref:CH&lt;x&gt;:FREQuency</a>	Sets or returns the Phase Reference Frequency
<a href="#">PHAseref:CH&lt;x&gt;:RANge?</a>	Returns the Phase Reference module guaranteed frequency range
<a href="#">PHAseref:CH&lt;x&gt;:STAtus?</a>	Returns the current Phase Reference Characterization status
<a href="#">PHAseref:MODe</a>	Sets or queries the Phase Correction Mode: OFF, FREerun, TRIGger, or SSCTrigger
<a href="#">PHAseref:SOUrce</a>	Sets or returns the Phase Reference source: C1 & C2, C3 & C4, C5 & C6, or C7 & C8

## Save and Recall Command Group

You use the commands in the Save and Recall Command Group to store and retrieve internal waveforms and settings. When you save a setup, you save all the settings of the instrument. When you recall a saved setting, the instrument restores itself to the state that it was in when you originally saved that setting.

Command	Description
<a href="#">DELEte:WAVEform</a>	Deletes (one or all) of the stored reference waveforms from memory
<a href="#">FACTory</a>	Resets the instrument to factory default settings
<a href="#">EXPort</a>	Exports waveform and histogram data to a file
<a href="#">EXPort:TYPE</a>	Sets or queries the form of the exported waveform data
<a href="#">IMPort:CUSTommask</a>	Imports and loads the custom mask
<a href="#">RECAll:SETUp</a>	Recalls saved instrument settings
<a href="#">RECAll:WAVEform</a>	Recalls a stored waveform into reference location
<a href="#">SAVe:SETUp</a>	Saves the current instrument settings to a specified location
<a href="#">SAVe:WAVEform</a>	Saves waveform in reference or file

## Status and Error Command Group

You use the commands in the Status and Error command Group to determine the status of the instrument and control events. Several commands and queries used with the instrument are common to all devices on the GPIB bus. The IEEE Std 488.2–1987 defines these commands and queries. The common commands begin with an asterisk (\*) character.

Command	Description
*CLS	Clears status
*ESE	Sets or returns the standard Event Status Enable Register
*ESR?	Returns the standard Event Status Register
*OPC	Sets OPC event when all pending operations are finished Returns "1" when all current operations complete
*OPT?	Returns a list of installed options
*PSC	Sets or returns the power on status flag
*RST	Resets the instrument to factory default settings
*SRE	Sets or returns the bits in the Service Request Enable Register
*STB?	Returns the contents of the Status Byte Register
*WAI	Prevents the instrument from executing further commands until all pending operations finish
ALLEv?	Returns all events
BUSY?	Returns instrument status
DESE	Sets or returns the bits in the Device Event Status Enable Register
EVENT?	Returns event code from the event queue
EVMsg?	Returns event code, message from the event queue
EVQty?	Returns number of events in the event queue
ID?	Returns identifying information about the instrument and its firmware

## System Command Group

You use the commands in the System Command Group to obtain information about your system, such as the serial numbers of your instrument, installed

modules, and the attached probes, the hardware version of the acquisition and processor circuit boards, and the gains and impedances of attached probes.

Command	Description
<a href="#">SYSTem:PROPERTIES:CH&lt;x&gt;:BANDwidth?</a>	Returns a list of available bandwidth selections for the specified channel (optical modules); for electrical modules, returns bandwidth characteristic if applicable
<a href="#">SYSTem:PROPERTIES:CH&lt;x&gt;:CAPacitance?</a>	Where applicable, returns sampling module load capacitance of the specified channel
<a href="#">SYSTem:PROPERTIES:CH&lt;x&gt;:CLKRec?</a>	Returns list of available clock recovery selections for the specified channel
<a href="#">SYSTem:PROPERTIES:CH&lt;x&gt;:CLKUser?</a>	Returns the allowable range of user clock-recovery rates for the module containing CH<x>
<a href="#">SYSTem:PROPERTIES:CH&lt;x&gt;:DYNamic?</a>	Returns sampling module lower and upper dynamic range limits of the specified channel
<a href="#">SYSTem:PROPERTIES:CH&lt;x&gt;:EXTender?</a>	Returns extender cable type of the specified channel
<a href="#">SYSTem:PROPERTIES:CH&lt;x&gt;:FILTer?</a>	Returns a list of available filter selections for the specified channel
<a href="#">SYSTem:PROPERTIES:CH&lt;x&gt;:IMPedance?</a>	Where applicable, returns sampling module input impedance of the specified channel
<a href="#">SYSTem:PROPERTIES:CH&lt;x&gt;:MODEInum?</a>	Returns sampling module model number per specified channel
<a href="#">SYSTem:PROPERTIES:CH&lt;x&gt;:NONDestruct?</a>	Where applicable, returns sampling module lower and upper maximum nondestructive range of the specified channel
<a href="#">SYSTem:PROPERTIES:CH&lt;x&gt;:OPERating?</a>	Where applicable, returns sampling module lower and upper operating range of specified channel
<a href="#">SYSTem:PROPERTIES:CH&lt;x&gt;:PRAnge?</a>	Queries the Phase Reference module frequency range, returning it as part of the system properties query results
<a href="#">SYSTem:PROPERTIES:CH&lt;x&gt;:PRObe: DYNamic?</a>	Where applicable, returns the probe lower and upper dynamic range limits of the specified channel
<a href="#">SYSTem:PROPERTIES:CH&lt;x&gt;:PRObe: IMPedance?</a>	Returns impedance of probe attached to specified channel
<a href="#">SYSTem:PROPERTIES:CH&lt;x&gt;:PRObe: MODEInum?</a>	Returns probe model number of the probe attached to the specified channel
<a href="#">SYSTem:PROPERTIES:CH&lt;x&gt;:PRObe: SCALE?</a>	Returns the probe scale factor of the probe attached to the specified channel
<a href="#">SYSTem:PROPERTIES:CH&lt;x&gt;:PRObe: SERIALnum?</a>	Returns serial number of probe attached to specified channel

Command	Description
<a href="#">SYSTem:PROPerTies:CH&lt;x&gt;:RISetime?</a>	Where applicable, returns the sampling module risetime characteristic of the specified channel
<a href="#">SYSTem:PROPerTies:CH&lt;x&gt;:SERialnum?</a>	Returns sampling module serial
<a href="#">SYSTem:PROPerTies:CH&lt;x&gt;:TEKPDriver?</a>	Returns sampling module TEKPROBE driver revision number of specified channel
<a href="#">SYSTem:PROPerTies:CH&lt;x&gt;:TEKPVersion?</a>	Returns sampling module TEKPROBE version number per channel
<a href="#">SYSTem:PROPerTies:CH&lt;x&gt;:WLENgth?</a>	Returns list of available wavelength selections for the specified channel
<a href="#">SYSTem:PROPerTies:EFEHWver?</a>	Returns the version number of the Electrical Front End (EFE) circuit board.
<a href="#">SYSTem:PROPerTies:EFESERialnum?</a>	Returns the serial number of the Electrical Front End (EFE) circuit board.
<a href="#">SYSTem:PROPerTies:GROup&lt;x&gt;:TEKPVersion?</a>	Returns the version number of the specified TEKPROBE controller
<a href="#">SYSTem:PROPerTies:MAInframe:KEY?</a>	Returns mainframe option key
<a href="#">SYSTem:PROPerTies:MAInframe:MODEInum?</a>	Returns mainframe model number
<a href="#">SYSTem:PROPerTies:MAInframe:SERialnum?</a>	Returns mainframe serial number
<a href="#">SYSTem:PROPerTies:MAInframe:SWVersion?</a>	Returns mainframe software version number
<a href="#">SYSTem:PROPerTies:MAInframe:UNIQUEid?</a>	Returns mainframe id number
<a href="#">SYSTem:PROPerTies:OFEHWver?</a>	Returns the Optical Front-End (OFE) board version number.
<a href="#">SYSTem:PROPerTies:OFESERialnum?</a>	Returns the serial number of the Optical Front End (OFE) circuit board.
<a href="#">SYSTem:PROPerTies:ONTime?</a>	Returns on time for current power up in hours
<a href="#">SYSTem:PROPerTies:POWerups?</a>	Returns number of power ups
<a href="#">SYSTem:PROPerTies:PROCHWver?</a>	Returns hardware version of Processor circuit board
<a href="#">SYSTem:PROPerTies:TBHWver?</a>	Returns the version of the Timebase (TB) circuit board in the instrument.
<a href="#">SYSTem:PROPerTies:TBSERialnum?</a>	Returns the serial number of the Timebase (TB) circuit board.
<a href="#">SYSTem:PROPerTies:TOTALontime?</a>	Returns the total accumulated on time in hours (across all power cycles)

## TDR Command Group

You use the commands in the TDR Command Group to do the following:

- Specify TDR step channels (C1 through C8), and select the polarity for each.
- Select TDR steps.
- Enable and disable acquisition.
- Set the TDR internal clock rate.
- Preset any TDR channel.
- Set the channel TDR deskew.
- Select the channel TDR units.
- Obtain all the TDR parameters.
- Preset TDR channel pairs for differential TDR measurements.
- Position the TDR Incident Edge.

Command	Description
<a href="#">AUTOSet:TDR:INCident</a>	Sets or returns the Incident edge option for the TDR autose (on or off)
<a href="#">TDR?</a>	Returns all TDR parameters
<a href="#">TDR:CH&lt;x&gt;:DIFpreset</a>	Sets the instrument to predefined differential TDR settings for the specified TDR channel pair
<a href="#">TDR:EXT10MHZref:FREQ</a>	Sets or returns the external 10 MHz reference frequency used for TDR acquisitions
<a href="#">TDR:CH&lt;x&gt;:PRESET</a>	Sets the instrument to predefined TDR settings for the specified TDR channel
<a href="#">TDR:REF10Mhz</a>	Sets or returns the source (internal or external) for the TDR 10 MHz reference frequency
<a href="#">TDR:CH&lt;x&gt;:STEP:DESkew</a>	Sets or returns the specified channel TDR deskew (if valid type head in channel)
<a href="#">TDR:CH&lt;x&gt;:STEP:POLarity</a>	Sets or returns the TDR step polarity for the specified channel
<a href="#">TDR:CH&lt;x&gt;:STEP:STATE</a>	Sets or returns whether the TDR generator is on or off for the specified channel
<a href="#">TDR:CH&lt;x&gt;:UNIts</a>	Sets or returns the TDR units for the specified channel
<a href="#">TDR:INTRate</a>	Sets or returns the TDR (Internal Clock) rate; same as TRIG:INTRATE

## Trigger Command Group

You use the commands in the Trigger Command Group to control all aspects of triggering for the instrument. You use the commands to do the following:

- Set the trigger source for triggering acquisition. You can select from the following sources:
  - Clock Sources:
    - Clock Input/Prescale Trigger (front panel input).
    - Internal clock recovery sources.
  - TDR Clock.
  - Trigger Direct Input (front panel input).
  - Free Run.
- For Clock Sources you can select one of three acquisition modes (Eye, Pattern, or Other).
- For Internal clock recovery sources you can set and query the clock recovery data rate.
- For TDR triggering you can set the trigger clock rate (25 - 300 kHz).
- For Trigger Direct source you can do the following:
  - Set the signal level and slope (rising or falling) of the trigger.
  - Set the trigger level to 50% of the trigger signal amplitude.
  - Set high frequency triggering hysteresis for improved triggering in specific instances.
- Set trigger holdoff, which is the time the instrument waits before arming the trigger system to accept triggers. Holdoff can help achieve stable triggering.
- Obtain all trigger parameters.
- Obtain the trigger system status (triggered, not triggered, or stopped).
- Set and query Pattern Autosync parameters (Pattern Length, Data Rate, Data to Clock Ratio).
- Initiate a Pattern Autosync.
- Control and Query USB-attached clock recovery instruments (such as the Tektronix CR125A, CR175A, or CR286A)

Command	Description
AUTOSet:TRIGger	Sets whether the trigger Autoset options are included when autoset executes Returns the status for the trigger Autoset options
TRIGger?	Returns all trigger parameters
TRIGger:AUTOSync	Autosyncs to the currently selected waveform, those pattern sync settings that have their AutoSync options set
TRIGger:AUTOSync:DATARate	Sets or returns the AutoSync option for data rate selection (on or off)
TRIGger:AUTOSync:DCRAtio	Sets or returns the AutoSync option for data-to-clock ratio selection (on or off)
TRIGger:AUTOSync:PLENght	Sets or returns the AutoSync option for pattern length selection (on or off)
TRIGger:CH<x>:CLKRec:LIST?	Returns a list of available clock recovery selections for the module containing CH<x>
TRIGger:CH<x>:CLKRec:RANge	Returns the allowable range of user clock recovery rates for the module containing CH<x>
TRIGger:CH<x>:CLKRec:USEr	Sets a user (custom) clock recovery rate to be a specified value within an allowable range, which is module-dependent Returns the rate currently defined as the user clock recovery rate
TRIGger:CH<x>:CLKRec:VALue	Sets or returns the clock recovery value for the specified channel
TRIGger:CLKRec:CRC:AUTOSConfigdev	Sets or queries the setup auto-save feature of the connected Clock Recovery instrument
TRIGger:CLKRec:CRC:CAPabilities?	Returns the instrument clock recovery device capabilities.
TRIGger:CLKRec:CRC:CLOCKAmplitude	Sets or queries the clock output amplitude of the connected Clock Recovery instrument
TRIGger:CLKRec:CRC:CLOCKOutput	Sets or queries the clock output mode of the connected Clock Recovery instrument
TRIGger:CLKRec:CRC:CREATESTandard	Saves a new clock recovery standard to the non-volatile memory of the connected Clock Recovery instrument
TRIGger:CLKRec:CRC:DATARate?	Returns the measured data rate (in Hz) from the connected Clock Recovery instrument
TRIGger:CLKRec:CRC:DELETESStandard	Deletes the named standard from the connected Clock Recovery instrument
TRIGger:CLKRec:CRC:DEVICENAME	Sets or queries the device name of the connected Clock Recovery instrument



Command	Description
TRIGger:CLKRec:CRC:EDGEDensity?	Returns the measured edge density (in percent) from the connected Clock Recovery instrument
TRIGger:CLKRec:CRC:EDGEDENSITYMode	Sets or queries the edge density mode setting of the connected Clock Recovery instrument
TRIGger:CLKRec:CRC:EQAlizer	Sets or queries the clock recovery instrument equalizer value.
TRIGger:CLKRec:CRC:EXPrev?	Returns the expansion board revision string of the connected Clock Recovery instrument
TRIGger:CLKRec:CRC:FPGarev?	Returns the FPGA revision string of the connected Clock Recovery instrument
TRIGger:CLKRec:CRC:HALFRate?	Returns the clock recovery device clock rate.
TRIGger:CLKRec:CRC:HWRRev?	Returns the hardware revision string of the connected Clock Recovery instrument
TRIGger:CLKRec:CRC:LOCKCount?	Returns the number of times the connected Clock Recovery instrument has relocked since the last reset
TRIGger:CLKRec:CRC:LOCKFp	Locks or unlocks the clock recovery instrument front panel controls.
TRIGger:CLKRec:CRC:LOCKMode	Sets or queries the lock mode of the connected Clock Recovery instrument
TRIGger:CLKRec:CRC:LOCKRange	Sets or queries the locking range about the nominal frequency setting of the connected Clock Recovery instrument
TRIGger:CLKRec:CRC:LOCKState?	Returns the lock state of the connected Clock Recovery instrument
TRIGger:CLKRec:CRC:LOOPBandwidth	Sets or queries the loop bandwidth of the connected Clock Recovery instrument
TRIGger:CLKRec:CRC:MODEInum?	Returns a string with the clock recovery instrument identifier information.
TRIGger:CLKRec:CRC:NOMEDgedensity	Sets or queries the nominal edge density of the connected Clock Recovery instrument
TRIGger:CLKRec:CRC:NOMFrequency	Sets or queries the nominal frequency setting of the connected Clock Recovery instrument
TRIGger:CLKRec:CRC:PEAKing	Sets or queries the peak range of the connected Clock Recovery instrument
TRIGger:CLKRec:CRC:PHASEERRLimit	Sets or queries the phase error limit of the connected Clock Recovery instrument
TRIGger:CLKRec:CRC:PHASEERRP2P?	Returns the phase error peak-to-peak measurement from the connected Clock Recovery instrument

Command	Description
<a href="#">TRIGger:CLKRec:CRC:PHASEERRRMS?</a>	Returns the phase error RMS measurement from the connected Clock Recovery instrument
<a href="#">TRIGger:CLKRec:CRC:RCONfigdevice</a>	Restores the connected Clock Recovery instrument to the settings from one of the non-volatile setup locations in the Clock Recovery Instrument
<a href="#">TRIGger:CLKRec:CRC:RELock</a>	Causes the connected Clock Recovery instrument to search for the data rate
<a href="#">TRIGger:CLKRec:CRC:RESETLockcount</a>	Resets the lock count to zero on the connected Clock Recovery instrument
<a href="#">TRIGger:CLKRec:CRC:SCONfigdevice</a>	Saves the current settings of the connected Clock Recovery instrument to one of the non-volatile setup locations in the Clock Recovery Instrument
<a href="#">TRIGger:CLKRec:CRC:SERIalnum?</a>	Returns the serial number of the connected Clock Recovery Instrument
<a href="#">TRIGger:CLKRec:CRC:STANdard</a>	Sets or queries the communications standard of the connected Clock Recovery instrument
<a href="#">TRIGger:CLKRec:CRC:STANDARDList?</a>	Returns a comma-separated list of the standards available on the connected Clock Recovery instrument
<a href="#">TRIGger:CLKRec:CRC:SUBCLOCKAmpl</a>	Sets or queries the sub-rate clock output amplitude of the connected Clock Recovery instrument
<a href="#">TRIGger:CLKRec:CRC:SUBCLOCKDiv</a>	Sets or queries the division ratio of the sub-rate clock of the connected Clock Recovery instrument
<a href="#">TRIGger:CLKRec:CRC:SUBCLOCKDIVList?</a>	Returns a comma-separated list of available sub-rate clock divisors of the connected Clock Recovery instrument
<a href="#">TRIGger:CLKRec:CRC:SUBCLOCKOutput</a>	Sets or queries the state of the sub-rate clock output of the connected Clock Recovery instrument
<a href="#">TRIGger:CLKRec:CRC:SWRev?</a>	Returns the firmware revision string of the connected Clock Recovery instrument
<a href="#">TRIGger:CLKRec:SOUrce</a>	Sets or returns the clock recovery channel source
<a href="#">TRIGger:HIFreq</a>	Sets or returns the high-frequency hysteresis
<a href="#">TRIGger:HOLDoff</a>	Sets or returns the trigger holdoff time
<a href="#">TRIGger:INTRate</a>	Sets or returns the TDR (Internal Clock) rate; same as TDR:INTRATE
<a href="#">TRIGger:LEVel</a>	Sets or returns the trigger level

Command	Description
<a href="#">TRIGger:SETLevel</a>	Sets the trigger level to 50% of the applied signal
<a href="#">TRIGger:SLOpe</a>	Sets or returns the trigger slope
<a href="#">TRIGger:SOUrce</a>	Sets or returns the trigger source
<a href="#">TRIGger:STATE?</a>	Returns the trigger system status

## Vertical Command Group

You use the commands in the Vertical Command Group to control the vertical setup of all live (channel) waveforms for acquisition and to control the display of channel, reference, and math waveforms. This group contains commands to set up the vertical parameters for either electrical or optical signals, depending on the sampling module in use.

You can use the commands in this group to do the following:

- Enable or disable the display of any channel, math, or reference waveform.
- Set which waveform is affected by the front-panel controls.
- Set the vertical scale for each channel, math, or reference waveform.
- Set the vertical position of a waveform. The vertical position affects only the display position of a channel, math, or reference waveform.
- Set the offset value to shift the vertical acquisition window to match the waveform data that you want to acquire (affects only channels).
- Set deskew and/or delay values to time align channels.
- Set the units for channel waveforms. If you set this value to Auto, the instrument automatically selects the units. If you set this value to any setting other than Auto (Volt, Amp, or Watt), the instrument will apply these units to the vertical axis.
- Set the external attenuation factor (as a multiplier or as a dB value) on an acquisition channel waveform to a value matching the amount of attenuation applied externally, before the signal enters the sampling head or probe tip input.
- Set the wavelength to match the wavelength of the optical signal you are using.
- Set the type of filtering, if any, you want performed on the selected optical channel.
- Specify bandwidth for use on the selected optical channel.

Command	Description
AUTOSet:VERTical	Sets or returns whether the vertical Autoset options are included when Autoset executes
CH<x>?	Returns the vertical parameters
CH<x>:ERCAL?	Returns whether calibrated extinction ratio is available for the optical module for the specified channel
CH<x>:BANdwidth:LIST?	Returns the bandwidths available for the specified channel
CH<x>:BANdwidth:VALue	Sets or returns the bandwidth for the specified channel
CH<x>:DELay	Sets or returns the delay time for the specified channel
CH<x>:DESkew	Sets or returns the deskew time
CH<x>:EXTAtten:MODE	Sets or returns the external attenuation mode
CH<x>:EXTAtten:VALue	Sets or returns the external attenuation factor (as a multiple)
CH<x>:FILTer:LIST?	Returns filter list for the specified channel
CH<x>:FILTer:VALue	Sets the hardware filter for the specified channel
CH<x>:OFFSet	Sets or returns the channel offset
CH<x>:POSition	Sets or returns the channel vertical position
CH<x>:SCAle	Sets or returns the channel vertical scale (per div)
CH<x>:UNIts	Sets or returns the channel vertical units
CH<x>:WFMLabel	Sets or returns the label associated with the specified channel waveform
CH<x>:WLENgth:LIST?	Returns the wavelengths available for the specified channel
CH<x>:WLENgth:VALue	Sets or returns the wavelength for the specified channel
REF<x>:POSition	Sets or returns the reference vertical position
REF<x>:SCAle	Sets or returns the reference vertical scale (per div)
REF<x>:WFMLabel	Sets or returns the label associated with the specified reference waveform
SElect?	Returns information on which waveforms are ON/OFF and which waveform is the selected waveform
SElect:CH<x>	Sets or returns a waveform (on or off) (main timebase)

Command	Description
<code>SElect:CONTROI</code>	Sets the waveform controlled by front panel. Note: NONE may be returned by *LRN?. If this is an argument, do nothing. Also, timebase is NOT optional  Returns the waveform and timebase selected for front-panel control
<code>SElect:MATH&lt;x&gt;</code>	Sets or returns a math waveform (on or off) (main timebase)
<code>SElect:REF&lt;x&gt;</code>	Sets or returns a reference waveform (on or off) (main timebase)

## Waveform Database Command Group

You use the commands in the Waveform Database Command group to view and set up a waveform database. A waveform database is a collection of sequentially acquired waveforms. Think of a waveform database as a three-dimensional array with a count dimension in addition to the usual vertical and horizontal dimensions. The count represents the number of times a specific waveform point has been acquired. Waveform database accumulation is always a dot mode accumulation (that is, no interpolation or vectoring is performed).

Waveform database count values stored in the waveform database array are unit-less with respect to absolute user units (for example, volts or seconds). They are described by the attributes of the source waveform along with the overall dimensions of the array.

The following are the dimensions of a waveform database:

- Horizontal (columns). Value is 1000, which is the maximum horizontal graticule view size.
- Vertical (rows). Value is 402, which is the maximum vertical graticule view size in pixels, plus one row each for the overrange (OR) and the underrange (UR) counts for each column.
- Count (weights or density). Value is 32 bits.

Use waveform databases for measurements, histogram calculations and display, mask testing, and generating a density-style graded display. If a waveform database is turned on for the selected source, then the mask testing, histogram calculations, and measurements can use the waveform database.

You can use the commands in this group to do the following:

- Set the source to which the waveform database is assigned.
- Enable or disable the waveform database.
- Set whether the waveform database is displayed.
- Set whether you want the waveform database drawn with color or intensity grading.
- Set the display emphasis for the waveform database. The value you specify sets the range of counts you want emphasized. The lowest value, 0, emphasizes bins with low counts; the highest value, 100, emphasizes bins with high counts.
- Invert the mapping function that applies colors and intensity to the waveform database to significantly to emphasize the least occurring pixels.
- Clear the specified waveform database.
- Obtain all the waveform database display parameters.
- Obtain all the waveform database parameters.

Command	Description
<a href="#">WFMDB?</a>	Returns all WfmDB parameters
<a href="#">WFMDB:DISPlay?</a>	Returns wWmDB display parameters
<a href="#">WFMDB:DISPlay:EMPHasis</a>	Sets or returns the WfmDB display emphasis
<a href="#">WFMDB:DISPlay:GRADing</a>	Sets or returns the WfmDB grading type
<a href="#">WFMDB:DISPlay:GRADMethod</a>	Sets or returns the WfmDB grading method to any one of four available methods
<a href="#">WFMDB:DISPlay:INVert</a>	Sets or returns the WfmDB grading inversion
<a href="#">WFMDB:WFMDB&lt;x&gt;?</a>	Returns WfmDB<x> parameters
<a href="#">WFMDB:WFMDB&lt;x&gt;:CLEar</a>	Deletes (clears) resource of specified WfmDB
<a href="#">WFMDB:WFMDB&lt;x&gt;:DISPlay</a>	Sets or returns the WfmDB<x> displayed
<a href="#">WFMDB:WFMDB&lt;x&gt;:ENABLE</a>	Sets or returns whether the specified waveform database is enabled
<a href="#">WFMDB:WFMDB&lt;x&gt;:PERSistence:COUNT</a>	Sets or returns the waveform count applied to WfmDB<x> when set to Variable Persistence mode
<a href="#">WFMDB:WFMDB&lt;x&gt;:PERSistence:MODE</a>	Sets or returns the persistence mode (Infinite or Variable) used by WfmDB<x>
<a href="#">WFMDB:WFMDB&lt;x&gt;:PERSistence:SAMPles?</a>	Returns the samples count in effect for WfmDB<x> when using in Variable Persistence mode
<a href="#">WFMDB:WFMDB&lt;x&gt;:SOURce</a>	Sets or returns the WfmDB<x> source

## Waveform Transfer Command Group

Use the commands in the Waveform Transfer Command Group to transfer waveform data points to and from the instrument. Waveform data points are a collection of values that define a waveform. One data value usually represents one data point in the waveform record. When working with envelope waveforms, each data value is either the minimum or the maximum of a min/max pair.

Before you transfer waveform data, you must specify the data format, record length, and waveform source.

### Data Formats

Acquired waveform data uses 19 or more bits to represent each data point. The number of bits used depends on the acquisition mode specified when you acquired the data. Data acquired in SAMple or ENVELOpe mode uses 19 bits per waveform data point. Data acquired in AVERage mode uses up to 24 bits per point.

The instrument can transfer waveform data in either ASCII or binary format. You specify the format with the DATA:ENCdg command. The instrument uses signed, 4 byte integers and floating point values; it does not support unsigned values.

ASCII data is represented by signed integer or floating point values. An example of an ASCII waveform data string may look like this:

```
:CURVE -509476864,-512163840,-511115264,-510918656,
-516161536,-484179968,-117112832,207093760,
322437120,351600640,409206784,468451328,
482934784,490668032,495517696,432668672,
482148352,483655680,493617152,495321088,
499843072,493420544,488964096,497221632, ...
```

ASCII will produce more readable and more easily formatted output than that produced by binary format. However, ASCII may require more bytes to send the same values than it does with binary. This may reduce transmission speeds.

Binary data can be represented by signed integer or floating-point values. The defined binary formats specify the order in which the bytes are transferred. The following are the four binary formats:

- RIBinary specifies signed integer data-point representation with the most significant byte transferred first.
- SRIBinary is the same as RIBinary except that the byte order is reversed; the least significant byte is transferred first. This format is useful when transferring data to IBM-compatible PCs.
- FPBinary specifies floating-point data-point representation with the most significant byte transferred first.
- SFPBinary is the same as RFBinary except that the byte order is reversed; the least significant byte is transferred first. This format is useful when transferring data to IBM-compatible PCs.

### Waveform Data and Record Lengths

You can transfer multiple points for each waveform record. You can transfer a portion of the waveform or you can transfer the entire record. You can use the `DATA:START` and `DATA:STOP` commands to specify the first and last data points of the waveform record. The instrument supports record lengths of 50, 100, 250, 500, 1000, 2000, 4000, 8000, and 16000 samples. The maximum record length for mag view waveforms is 4000 points.

When transferring data into the instrument, you must first specify the record length of the destination waveform record. To specify the record length, use the `WFMinpre:NR_Pt` command. Next, specify the first data point within the waveform record. For example, when you set `DATA:START` to 1, data points will be stored starting with the first point in the record. The instrument will ignore the value set by `DATA:STOP` when reading in data. It will stop reading in data when there is no more data to read, or when it has reached the specified record length.

When transferring data from the instrument, you must specify the first and last data points in the waveform record. Setting `DATA:START` to 1 and `DATA:STOP` to the record length will always return the entire waveform.

### Waveform Data Locations and Memory Allocation

The `DATA:SOURce` and `DATA:TIMEbase` commands specify the waveform source and waveform timebase when transferring a waveform from the instrument. You can only transfer one waveform at a time.

Waveforms sent to the instrument are always stored in one of the eight reference memory locations. You use the `DATA:DESTination` command to specify a reference memory location.

### Waveform Preamble

Each waveform that you transfer has an associated waveform preamble that contains information such as the horizontal scale, the vertical scale, and other settings in effect when the waveform was created. The values returned by the `WFMOutpre` query commands are set by the previous curve query. Therefore, to have a valid set of preamble and data, you must perform a `CURVe?` query followed by a `WFMOutpre?` query. Refer to the individual `WFMinpre` and `WFMOutpre` commands for more information.



Command	Description
<a href="#">CURVe</a>	The command form transfers waveform data to the instrument (reference memory location specified by DATA:DESTination) The query form transfers waveform data from the instrument specified by the DATA:SOURce command
<a href="#">DATA</a>	Sets the format and location of the waveform data that is transferred with the CURVe command Returns the format and location of the waveform data that is transferred with the CURVe? command
<a href="#">DATA:DESTination</a>	Sets or returns the reference memory location for storing waveform data sent to the instrument
<a href="#">DATA:ENCdg</a>	Sets or returns the format of the waveform data
<a href="#">DATA:SOURce</a>	Sets or returns the location of waveform data transferred from the instrument
<a href="#">DATA:START</a>	Sets or returns the starting data point in waveform transfer
<a href="#">DATA:STOP</a>	Sets or returns the ending data point in waveform transfer
<a href="#">WAVFrm?</a>	Returns WFMOutpre? and CURVe? data for the waveform as specified by the DATA:SOURce command
<a href="#">DATA:TIMEbase</a>	Sets or returns the DATA:SOURce associated timebase used to specify the location of the waveform data to be transferred from the instrument
<a href="#">WFMInpre?</a>	Returns the waveform formatting data (preamble) for the incoming waveform
<a href="#">WFMInpre:BIT_Nr?</a>	Returns the number of bits (width) per waveform point for the waveform to be transferred to the instrument
<a href="#">WFMInpre:BN_Fmt</a>	Sets or returns the format of the binary or ascii data encoding of the waveform to be transferred to the instrument
<a href="#">WFMInpre:BYT_Nr?</a>	Returns the byte width of the waveform to be transferred to the instrument
<a href="#">WFMInpre:BYT_Or</a>	Sets or returns which byte of binary waveform data is transmitted first during a waveform data transfer to the instrument

<b>Command</b>	<b>Description</b>
<a href="#">WFMInpre:ENCdg</a>	Sets or returns the type of encoding used for waveform data transferred with the CURVe command
<a href="#">WFMInpre:NR_Pt</a>	Sets or returns the record length of the input waveform
<a href="#">WFMInpre:PT_Fmt</a>	Sets or returns the data point format of the incoming waveform
<a href="#">WFMInpre:WFMLabel</a>	Sets or returns the label for the incoming waveform
<a href="#">WFMInpre:XINcr</a>	Sets or returns the interval between samples of the incoming waveform
<a href="#">WFMInpre:XMUlt</a>	Sets or returns the horizontal scale factor (horizontal units per division) of the incoming waveform
<a href="#">WFMInpre:XOFF</a>	Sets or returns the horizontal position in horizontal units of the incoming waveform
<a href="#">WFMInpre:XUNit</a>	Sets or returns the horizontal (x-axis) units of the incoming waveform
<a href="#">WFMInpre:XZEro</a>	Sets or returns the horizontal (x-axis) origin offset (time of first point in this instrument) of the incoming waveform
<a href="#">WFMInpre:YMUlt</a>	Sets or returns the vertical scale in vertical units per division of the incoming waveform returned by the previous CURVe? command
<a href="#">WFMInpre:YOFF</a>	Sets or returns the vertical position of the incoming waveform in divisions
<a href="#">WFMInpre:YSCALE</a>	Sets or returns the vertical unit scale factor of the incoming waveform
<a href="#">WFMInpre:YUNit</a>	Sets or returns the vertical (y-axis) units of the incoming waveform
<a href="#">WFMInpre:YZEro</a>	Sets or returns the vertical offset voltage of the incoming waveform
<a href="#">WFMOutpre?</a>	Returns the preamble for the outgoing waveform
<a href="#">WFMOutpre:BIT_Nr?</a>	Returns the number of bits (width) per waveform point of the waveform to be transferred from the instrument
<a href="#">WFMOutpre:BN_Fmt</a>	Sets or returns the format of the binary data encoding of the waveform to be transferred from the instrument
<a href="#">WFMOutpre:BYT_Nr?</a>	Returns the byte width for the waveform to be transferred from the instrument

Command	Description
<a href="#">WFMOutpre:BYT_Or</a>	Sets or returns which byte of binary waveform data is transmitted first during a waveform data transfer
<a href="#">WFMOutpre:ENCdg</a>	Sets or returns the type of encoding used for waveform data transferred with the CURVe? command
<a href="#">WFMOutpre:NR_Pt?</a>	Returns the number of points that were returned by the previous CURVe? command
<a href="#">WFMOutpre:PT_Fmt?</a>	Returns the data point format of the waveform returned by the previous CURVe? command
<a href="#">WFMOutpre:WFId?</a>	Returns a descriptive string of the waveform returned by the previous CURVe? command
<a href="#">WFMOutpre:WFMLabel?</a>	Returns the label for the previous waveform transferred by the CURVe? command
<a href="#">WFMOutpre:XINcr?</a>	Returns the interval between samples of the waveform transferred by the CURVe? command
<a href="#">WFMOutpre:XMUIt?</a>	Returns the horizontal scale factor of the waveform returned by the previous CURVe? command
<a href="#">WFMOutpre:XOFF?</a>	Returns the horizontal position in horizontal units of the returned waveform
<a href="#">WFMOutpre:XUNit?</a>	Returns the horizontal units of the waveform returned by the previous CURVe? command
<a href="#">WFMOutpre:XZEro?</a>	Returns the time of first point (sample) of the waveform returned by the previous CURVe? command
<a href="#">WFMOutpre:YMUIt?</a>	Returns the vertical scale factor in vertical units per division of the waveform returned by the previous CURVe? command
<a href="#">WFMOutpre:YOFF?</a>	Returns the vertical position of the waveform returned by the previous CURVe? command
<a href="#">WFMOutpre:YSCALE?</a>	Returns vertical unit scale factor of the incoming waveform by the previous CURVe? command
<a href="#">WFMOutpre:YUNit?</a>	Returns the vertical units of the waveform returned by the previous CURVe? command
<a href="#">WFMOutpre:YZEro?</a>	Returns the vertical offset of the waveform returned by the previous CURVe? command



---

# Commands Listed in Alphabetical Order

## ACQuire? (Query Only)

This is a query only command that returns all the current acquisition parameters.

**Group** Acquisition

**Syntax** ACQuire?

**Examples** ACQUIRE? might return the following string for the current acquisition parameters: ACQUIRE:CURRENTCOUNT:ACQWFMS 0;HISTWFMS 0;HISTHITS 0;MASKWFMS 0;MASKSAMPLES 0;MASKTHITS 0;MASKHITS1 0;MASKHITS2 0;MASKHITS3 0;MASKHITS4 0;MASKHITS5 0;MASKHITS6 0;MASKHITS7 0;MASKHITS8 0;:ACQUIRE:MODE SAMPLE;NUMAVG 16;SAVEFILE:SAVESCREEN "";SAVEWFM "";:ACQUIRE:STATE 1;STOPAFTER:MODE RUNSTOP;BELL 0;ACTION NONE;CONDITION ACQWFMS;COUNT 1.

## ACQuire:CURRentcount:ACQWfms? (Query Only)

This query only command returns the current count value of acquired waveforms. The target value of this count is set by the ACQuire:STOPAfter:COUNT command (in conjunction with the ACQuire:STOPAfter:CONDition command). The instrument then counts up to this value. When the count reaches (or exceeds) the value, acquisition stops, and the specified StopAfter action is enabled.

**Group** Acquisition

**Syntax** ACQuire:CURRentcount:ACQWfms?

**Related Commands** [ACQuire:STOPAfter:COUNT](#), [ACQuire:STOPAfter:CONDition](#)

**Returns** NR1 is the current count value of acquired waveforms.

**Examples** ACQUIRE:CURRENTCOUNT:ACQWFMS? might return ACQUIRE:CURRENTCOUNT:ACQWFMS 20, indicating that currently 20 waveforms have been acquired.

## ACQuire:CURRentcount:HISTHits? (Query Only)

This query only command returns the current count value of histogram hits. The target value of this count is set by the ACQuire:STOPAfter:COUNT command (in conjunction with the ACQuire:STOPAfter:CONDition command). The instrument then counts up to this value. When the count reaches (or exceeds) this value, acquisition stops, and the specified StopAfter action is enabled.

**Group** Acquisition

**Syntax** ACQuire:CURRentcount:HISTHits?

**Related Commands** [ACQuire:STOPAfter:COUNT](#), [ACQuire:STOPAfter:CONDition](#)

**Returns** NR1 is the current count value of histogram hits.

**Examples** ACQUIRE:CURRENTCOUNT:HISTHITS? might return  
 ACQUIRE:CURRENTCOUNT:HISTHITS 100, indicating that currently 100  
 histogram hits have been acquired.

## ACQuire:CURRentcount:HISTWfms? (Query Only)

This query only command returns the current count value of waveforms accumulated for histograms. The target value of this count is set by the ACQuire:STOPAfter:COUNT command (in conjunction with the ACQuire:STOPAfter:CONDition command). The instrument then counts up to this value. When the count reaches (or exceeds) this value, acquisition stops, and the specified StopAfter action is performed.

**Group** Acquisition

**Syntax** ACQuire:CURRentcount:HISTWfms?

**Related Commands** [ACQuire:STOPAfter:COUNT](#), [ACQuire:STOPAfter:CONDition](#)

**Returns** NR1 is the current count value of histogram waveforms.

**Examples** ACQUIRE:CURRENTCOUNT:HISTWFMS? might return  
ACQUIRE:CURRENTCOUNT:HISTWFMS 25, indicating that currently 25 waveforms have been acquired in the histogram.

## ACQUIRE:CURRENTCOUNT:MASKHITS<x>? (Query Only)

This query only command returns the current count value of hits in the specified mask polygon, which can be 1 through 8. The target value of this count is set by the ACQUIRE:STOPAFTER:COUNt command (in conjunction with the ACQUIRE:STOPAFTER:CONDition command). The instrument then counts up to this value. When the count reaches (or exceeds) this value, acquisition stops, and the specified StopAfter action is enabled.

**Group** Acquisition

**Syntax** ACQUIRE:CURRENTCOUNT:MASKHITS<x>?

**Related Commands** [ACQUIRE:STOPAFTER:COUNt](#), [ACQUIRE:STOPAFTER:CONDition](#)

**Returns** NR1 is the current count value of hits for the specified mask polygon, which can be 1 through 8.

**Examples** ACQUIRE:CURRENTCOUNT:MASKHITS4 might return  
ACQUIRE:CURRENTCOUNT:MASKHITS4 300, indicating that currently 300 mask hits have been acquired in mask 4.

## ACQUIRE:CURRENTCOUNT:MASKSAMPLES? (Query Only)

This query only command returns the current count value of mask samples. The target value of this count is set by the ACQUIRE:STOPAFTER:COUNt command (in conjunction with the ACQUIRE:STOPAFTER:CONDition command). The instrument then counts up to this value. When the count reaches (or exceeds) this value, acquisition stops, and the specified StopAfter action is performed.

**Group** Acquisition

**Syntax** ACQUIRE:CURRENTCOUNT:MASKSAMPLES?

**Related Commands** [ACQUIRE:STOPAFTER:COUNt](#), [ACQUIRE:STOPAFTER:CONDition](#)

**Returns** NR1 is the current count value of mask samples accumulated for mask testing.

**Examples** ACQUIRE:CURRENTCOUNT:MASKSAMPLES? might return  
ACQUIRE:CURRENTCOUNT:MASKSAMPLES 75, indicating that  
currently 75 mask samples have been acquired for mask testing.

## ACQUIRE:CURRENTCOUNT:MASKTHITS? (Query Only)

This query only command returns the current total count value of mask hits. The target value of this count is set by the ACQUIRE:STOPAFTER:COUNT command (in conjunction with the ACQUIRE:STOPAFTER:CONDITION command). The instrument then counts up to this value. When the count reaches (or exceeds) this value, acquisition stops, and the specified StopAfter action is performed.

**Group** Acquisition

**Syntax** ACQUIRE:CURRENTCOUNT:MASKTHITS?

**Related Commands** [ACQUIRE:STOPAFTER:COUNT](#), [ACQUIRE:STOPAFTER:CONDITION](#)

**Returns** NR1 is the current count value of mask hits.

**Examples** ACQUIRE:CURRENTCOUNT:MASKTHITS? might return  
ACQUIRE:CURRENTCOUNT:MASKTHITS 300, indicating that currently 300 mask  
hits have been acquired.

## ACQUIRE:CURRENTCOUNT:MASKWFMS? (Query Only)

This query only command returns the current count value of mask waveforms. The target value of this count is set by the ACQUIRE:STOPAFTER:COUNT command (in conjunction with the ACQUIRE:STOPAFTER:CONDITION command). The instrument then counts up to this value. When the count reaches (or exceeds) this value, acquisition stops, and the specified StopAfter action is performed.

**Group** Acquisition

**Syntax** ACQUIRE:CURRENTCOUNT:MASKWFMS?



<b>Related Commands</b>	<a href="#">ACQUIRE:STOPAfter:COUNT</a> , <a href="#">ACQUIRE:STOPAfter:CONDition</a>
<b>Returns</b>	NR1 is the current count value of waveforms accumulated for mask testing.
<b>Examples</b>	ACQUIRE:CURRENTCOUNT:MASKWFMS? might return ACQUIRE:CURRENTCOUNT:MASKWFMS 25, indicating that currently 25 waveforms have been acquired for mask testing.

## ACQUIRE:DATA:CLEAR (No Query Form)

This command (no query form) causes an acquisition reset and clears all acquired data, clears the display, and, if active, resets FrameScan. When a clear data occurs, it has the following effects:

- Sample-mode waveform. The current waveform data is replaced by the waveform data of the next acquisition cycle when it is available.
- Average-mode waveform. The average algorithm mode resets such that the next acquisition will be a straight-through copy.
- Envelope-mode waveform. The envelope algorithm mode resets such that the next acquisition will be a straight-through copy.
- Waveform database. The contents of all databases are cleared (reset to zero) immediately.
- Histogram data and statistics. The data and all statistics will be cleared immediately.
- Mask counts and statistics. Mask counts and all statistics will be cleared immediately.
- Acquisition status. The acquisition waveform count or StopAfter condition is reset and target counts are reloaded.
- Counts. Resets all counts, including number of acquired waveforms, acquisition and math average counts, conditional stop counts, and FrameScan counts (if any of these are active).
- Measurement statistics. Measurement statistics are reset.

**Group** Acquisition

**Syntax** ACQUIRE:DATA:CLEAR

**Examples** ACQUIRE:DATA:CLEAR causes an acquisition reset and clears all acquired data.

## ACQUIRE:MODE

This command sets or queries the acquisition mode of the instrument, which determines how the final value of the acquisition interval is generated from the many data samples. The instrument applies the specified mode globally to all channel waveforms that it acquires. Sending this command is equivalent to selecting the Acquisition Mode in the Acquisition Setup dialog box.

Normally, the instrument takes one sample per trigger and builds a channel waveform over many successive triggers. If acquiring continues, the instrument writes over the acquired waveform with each newly acquired waveform from the same channel. You can modify this behavior by setting the instrument to use one of the three, mutually exclusive acquisition modes:

- **Sample.** Use Sample mode to see the signal in its purest form with no post processing. This is the default mode.
- **Average.** Use Average mode to reduce the apparent noise in the signal to reveal fundamental waveform behavior.
- **Envelope.** Use Envelope mode when it's important to see the variation of extremes in a waveform over time.

**Group** Acquisition

**Syntax** ACQUIRE:MODE { SAMPLE | AVERAGE | ENVELOPE }  
ACQUIRE:MODE?

**Related Commands** [ACQUIRE:NUMAVG](#)

- Arguments**
- **SAMPle** specifies sample mode, in which the displayed data point value is simply the sampled value that was taken during the acquisition interval. There is no post processing of acquired samples; the instrument overwrites waveforms at each new acquisition cycle. **SAMPle** is the default acquisition mode.
  - **AVERAGE** specifies averaging mode, in which the resulting waveform shows an average of **SAMPle** data points from several consecutive waveform acquisitions. The instrument processes the number of waveforms you specify into the acquired waveform, creating a running back-weighted exponential average of the input signal. The number of waveform acquisitions that go into making up the average waveform is set or queried using the **ACQUIRE:NUMAVg** command.
  - **ENVELOpe** specifies envelope mode, in which the instrument continuously, as subsequent waveforms are acquired, retains the running minimum (Min) and maximum (Max) values in adjacent sample intervals, creating an envelope of all waveforms acquired for that channel.

**Examples** **ACQUIRE:MODE ENVELOPE** sets the acquisition mode to display a waveform that is an envelope of many individual waveform acquisitions.

**ACQUIRE:MODE?** might return **ACQUIRE:MODE AVERAGE**, indicating that the displayed waveform is the average of the specified number of waveform acquisitions.

## ACQUIRE:NUMAVg

This command sets or queries the number of waveform acquisitions that make up an averaged waveform. Sending this command is equivalent to setting the number of samples in the Acquisition Mode section of the Acquisition Setup dialog box when Average is checked for the Acquisition Mode. Use the **ACQUIRE:MODE** command to enable the Average mode.

**Group** Acquisition

**Syntax** **ACQUIRE:NUMAVg <NR1>**  
**ACQUIRE:NUMAVg?**

**Related Commands** [ACQUIRE:MODE](#)

**Arguments** **NR1** is the number of consecutive waveform acquisitions (from 2 to 4,096) used for averaging.

- Examples** ACQUIRE:NUMAVG 10 specifies that an averaged waveform will show the result of combining 10 separately acquired waveforms.
- ACQUIRE:NUMAVG? might return ACQUIRE:NUMAVG 75, indicating that there are 75 acquisitions specified for averaging.

## ACQUIRE:SAVEFILE:SAVESCREEN

This command sets or queries the file to which a screen is saved at the completion of a StopAfter condition (when the StopAfter action is set to SAVESCREEN). See the [ACQUIRE:STOPAFTER:ACTION](#) command. The screen is saved in the specified file in .BMP format. This is equivalent to selecting Print Screen to File in the Stop Action section of the Acquisition Setup dialog box and entering a file name.

**Group** Acquisition

**Syntax** ACQUIRE:SAVEFILE:SAVESCREEN <Qstring>  
ACQUIRE:SAVEFILE:SAVESCREEN?

**Related Commands** [ACQUIRE:STOPAFTER:ACTION](#), [ACQUIRE:STOPAFTER:CONDITION](#),  
[ACQUIRE:STOPAFTER:MODE](#)

**Arguments** <Qstring> defines the file name and path. Input the file path using the form <drive>/<dir>/<filename>. The <drive> and one or more <dir> arguments are optional. If you do not specify them, the instrument will copy the file into the current directory. The <filename> can be a Windows long file name. Do not use wild card characters.

**Examples** ACQUIRE:SAVEFILE:SAVESCREEN "wavetest.bmp" specifies the file Wavetest.bmp (in the current directory) to which the screen will be saved after a StopAfter SAVESCREEN condition is met.

ACQUIRE:SAVEFILE:SAVESCREEN? might return  
ACQUIRE:SAVEFILE:SAVESCREEN "TESTFILE.BMP", indicating that the screen will be saved in TESTFILE.BMP in the current directory when the StopAfter action is enabled.

## ACQUIRE:SAVEFILE:SAVEWFM

This command sets or queries the file to which waveform data is saved at the completion of a StopAfter condition (when the StopAfter action is set to SAVEWAVEFORM). See the [ACQUIRE:STOPAFTER:ACTION](#) command. This is equivalent to selecting Save All Waveforms in the Stop Action section of the

Acquisition Setup dialog box and entering a file name. This command sets the base file name and path used when multiple waveforms are saved. Each saved waveform has the base file name with a unique index number appended to it to prevent overwriting of the files.

**Group** Acquisition

**Syntax** ACQUIRE:SAVEFILE:SAVEWFM <Qstring>  
ACQUIRE:SAVEFILE:SAVEWFM?

**Related Commands** [ACQUIRE:STOPAfter:ACTion](#), [ACQUIRE:STOPAfter:CONDition](#),  
[ACQUIRE:STOPAfter:MODE](#)

**Arguments** <QSTRING> defines the base file name and path. Input the file path using the form <drive>/<dir>/<filename>. The <drive> and one or more <dir> arguments are optional. If you do not specify them, the instrument will copy the file into the current directory. The <filename> can be a Windows long file name. Do not use wild card characters. An index number is attached to the base filename each time this command is sent so that the file is not overwritten. For example, a file with the base name Wave.doc will be saved as Wave1.doc the first time, Wave2.doc the second time, and so on.

**Examples** ACQUIRE:SAVEFILE:SAVEWFM "wave.doc" specifies the file Wave1.doc (in the current directory) to which the screen will be saved after a StopAfter SAVEWFM condition is met.

ACQUIRE:SAVEFILE:SAVEWFM? might return ACQUIRE:SAVEFILE:SAVEWFM "WAVETST.DOC", indicating that the waveform data will be saved in WAVETST.DOC in the current directory when the StopAfter action is enabled.

## ACQUIRE:STATE

This command starts or stops acquisitions or queries whether the acquisition is running or stopped. Sending this command is equivalent to pressing the front-panel RUN/STOP button. The [ACQUIRE:STOPAfter:MODE](#) and [ACQUIRE:STOPAfter:CONDition](#) commands may cause other signal events to also stop acquisition.

**Group** Acquisition

**Syntax** ACQUIRE:STATE { OFF | ON | RUN | STOP | <NR1> }  
ACQUIRE:STATE?

**Related Commands** [ACQUIRE:STOPAfter:MODE](#), [ACQUIRE:STOPAfter:CONDition](#)

- Arguments**
- OFF stops acquisitions.
  - STOP stops acquisitions.
  - ON starts acquisitions.
  - RUN starts acquisitions.
  - NR1 set to 0 stops acquisitions; any other value starts acquisitions.

**Examples** ACQUIRE:STATE RUN starts acquisition of waveform data.  
ACQUIRE:STATE? might return ACQUIRE:STATE 1, indicating that the acquisition system is running.

---

**NOTE.** *A query always returns a 0 or 1.*

---

## ACQUIRE:STOPAfter? (Query Only)

This query only command returns all StopAfter parameters.

**Group** Acquisition

**Syntax** ACQUIRE:STOPAfter?

**Examples** ACQUIRE:STOPAFTER? might return ACQUIRE:STOPAFTER:MODE RUNSTOP;BELL 0;ACTION NONE;CONDITION ACQWFMS;COUNT 1.

## ACQUIRE:STOPAfter:ACTion

This command sets or queries the action that the instrument performs after the acquisition is stopped when reading a conditional state. This is equivalent to making a Stop Action selection in the Acquisition Setup dialog box.

**Group** Acquisition

**Syntax** ACQUIRE:STOPAfter:ACTion { NONE | SAVEScreen | PRINTScreen | SAVEwaveform }  
ACQUIRE:STOPAfter:ACTion?

**Related Commands** [ACQUIRE:SAVEFile:SAVEScreen](#), [ACQUIRE:SAVEFile:SAVEWfm](#)

- Arguments**
- **NONE** specifies to take no action when the stop after condition is met. This is the default.
  - **SAVEScreen** specifies that on acquisition stop the file named by the `ACQUIRE:SAVEFile:SAVEScreen` command will be created if it does not exist or overwritten if does. The resultant file will contain a bitmap image of the display.
  - **PRINTScreen** specifies that on acquisition stop a bitmap image of the application is sent to the default printer.
  - **SAVEWaveform** specifies that on acquisition stop all waveforms are saved in the files specified by the `ACQUIRE:SAVEFile:SAVEWfm` command.

**Examples**

`ACQUIRE:STOPAFTER:ACTION PRINTSCREEN` sets the instrument to send a bitmap image of the application to the printer when the stop after condition is met.

`ACQUIRE:STOPAFTER:ACTION?` might return `ACQUIRE:STOPAFTER:ACTION NONE`, indicating that no action is taken when the stop after condition is met.

## ACQUIRE:STOPAfter:BELL

This command sets or queries whether to sound the bell after the acquisition is complete. This is equivalent to checking Ring Bell in the Stop Action section of the Acquisition Setup dialog box.

**Group** Acquisition

**Syntax** `ACQUIRE:STOPAfter:BELL { ON | OFF | NR1 }`  
`ACQUIRE:STOPAfter:BELL?`

- Arguments**
- **ON** turns on the bell.
  - **OFF** turns off the bell.
  - **NR1** set to 0 turns off the bell; any other value turns on the bell.

**Examples**

`ACQUIRE:STOPAFTER:BELL ON` turns on the Stop After Bell feature.

`ACQUIRE:STOPAFTER:BELL?` might return `ACQUIRE:STOPAfter:BELL 1`, indicating that the Stop After Bell feature is on.

**NOTE.** *A query always returns a 0 or 1.*

---

**NOTE.** *The DSA8300 does not have an internal speaker. To hear a "bell" on stop, connect a speaker to the audio Line Output jack on the back of the instrument.*

---

## ACQuire:STOPAfter:CONDition

This command sets or queries the StopAfter condition. The StopAfter condition qualifies a stop condition for the acquisition system. Only one StopAfter condition can be active at a given time. Each StopAfter condition identifies, directly or indirectly, a specific data element or operation such that all mutually exclusive conditions are unique and unambiguous. This command allows you to specify the condition on which to stop acquiring. The condition is valid when the ACQuire:STOPAfter:MODE is set to CONDition. This is equivalent to checking the Condition control in the Stop After section of the Acquisition Setup dialog box and selecting a condition.

---

**NOTE.** *You must have the associated feature enabled before you can set the StopAfter condition. For example, you must turn on histograms (HISTogram:ENABle) before you can set the HISTHits StopAfter condition.*

---

**Group** Acquisition

**Syntax** ACQuire:STOPAfter:CONDition { ACQwfms | HISTwaveform | HISTHits | MASKwaveform | MASKSample | MASK<x>Hits | MASKTOTALhit | FRAMEcycle | AVGComp }  
ACQuire:STOPAfter:CONDition?

**Related Commands** [ACQuire:STOPAfter:COUNt](#), [ACQuire:STOPAfter:MODE](#)

- Arguments**
- ACQwfms sets the instrument to stop acquiring after some specified number of raw acquisition cycles. This setting tells the instrument to count the number of Main Timebase sweeps (Mag sweeps are not counted independently) and stop acquisition after the specified number of acquisitions has been reached. Use the ACQuire:STOPAfter:COUNt command to set the target number of waveforms.
  - HISTwaveform sets the instrument to stop acquisition after a specified number of Histogram source waveforms is acquired. Use the ACQuire:STOPAfter:COUNt command to set the specified number of waveforms.



- **HISTHits** sets the instrument to stop acquisition after it acquires a specified number of valid hits (that is, non-null samples) in the histogram region. This is a greater than or equal to condition because only complete records are processed. Therefore, the number of actual hits will not necessarily match your requested number of hits; the acquisition will stop when the number of hits is greater than or equal to the number of specified hits. Use the `ACQUIRE:STOPAFTER:COUNT` command to set the specified number of hits.
- **MASKWaveform** sets the instrument to stop acquisition after a specified number of the Mask source waveforms are acquired. Use the `ACQUIRE:STOPAFTER:COUNT` command to set the specified number of waveforms.
- **MASKSample** sets the instrument to stop acquisition after a greater than or equal to number of valid (that is, non-null) waveform samples are acquired of the Mask source waveform. Use the `ACQUIRE:STOPAFTER:COUNT` command to set the specified number of waveform samples.
- **MASK<x>Hits** sets the instrument to stop acquisition after a specified number of mask hits occur in mask number <x>, which can be 1 through 8. For the same reasons as HISTHits argument, this is also a greater than or equal to condition. Use the `ACQUIRE:STOPAFTER:COUNT` command to set the specified number of hits.
- **MASKTotalHit** sets the instrument to stop acquisition after a specified number of total mask hits occurs in all masks combined. For the same reasons as the HISTHits argument, this is also a greater than or equal to condition. Use the `ACQUIRE:STOPAFTER:COUNT` command to set the specified total number of hits.
- **FRAMEcycle** sets the instrument to stop acquisition after it has acquired the number of bits specified by the `HORIZONTAL:FRAMESCAN:SCANBITS` command.
- **AVGComp** sets the instrument to stop acquisition after the number of waveforms specified by the `ACQUIRE:NUMAVG` command have been acquired and averaged.

### Examples

`ACQUIRE:STOPAFTER:CONDITION HISTHits` sets the instrument to stop acquisition after it acquires the specified number of valid hits in the histogram region.

`ACQUIRE:STOPAFTER:CONDITION?` might return

`ACQUIRE:STOPAFTER:CONDition MASK3HITS`, indicating that the acquisition will be stopped after the hits in mask 3 are equal to or greater than the specified number of hits.

## ACQUIRE:STOPAFTER:COUNT

This command sets or queries the target StopAfter count for the condition specified by the ACQUIRE:STOPAFTER:CONDITION command. The current count for the condition must be equal to or greater than this value before acquisitions are stopped and a StopAfter action is enabled. The state of the numeric StopAfter count for each condition is kept individually so that you do not need to re-enter a count when switching between conditions. Use the appropriate ACQUIRE:CURRENTCOUNT command to get the current count for a condition (see *Related Commands* below).

**Group** Acquisition

**Syntax** ACQUIRE:STOPAFTER:COUNT <NR1>  
ACQUIRE:STOPAFTER:COUNT?

**Related Commands** [ACQUIRE:STOPAFTER:MODE](#), [ACQUIRE:STOPAFTER:CONDITION](#), [ACQUIRE:CURRENTCOUNT:ACQWFMS?](#), [ACQUIRE:CURRENTCOUNT:HISTWFMS?](#), [ACQUIRE:CURRENTCOUNT:HISTHITS?](#), [ACQUIRE:CURRENTCOUNT:MASKWFMS?](#), [ACQUIRE:CURRENTCOUNT:MASKHITS<x>?](#), [ACQUIRE:CURRENTCOUNT:MASKSAMPLES?](#)

**Arguments** NR1 is the count value that must be reached (or exceeded) before the acquisitions stop and StopAfter action can occur.

**Examples** ACQUIRE:STOPAFTER:COUNT 12 sets the StopAfter count for the specified condition to 12.

ACQUIRE:STOPAFTER:COUNT? might return ACQUIRE:STOPAFTER:COUNT 5, indicating that the total count for the specified condition is 5.

## ACQUIRE:STOPAFTER:MODE

This command tells the instrument when to stop taking acquisitions. The query form of this command returns the StopAfter mode. This command is the equivalent of checking Run/Stop or Condition in the Stop After section of the Acquisition Setup dialog box. (The ACQUIRE:STATE command can also be used to start or stop acquisitions.)

**Group** Acquisition

**Syntax** ACQUIRE:STOPAfter:MODE { RUNSTOP | CONDITION }  
ACQUIRE:STOPAfter:MODE?

**Related Commands** [ACQUIRE:STOPAfter:CONDITION](#), [ACQUIRE:STATE](#)

- Arguments**
- RUNSTOP specifies that the run and stop state is determined by the front-panel RUN/STOP controls.
  - CONDITION specifies that the run and stop state of the system is determined by a set of qualifiers specified by the StopAfter Condition. These sub-states are further described in the ACQUIRE:STOPAfter:CONDITION section. (The instrument can still be stopped unconditionally by pressing the front-panel Run/Stop button or by sending the ACQUIRE:STATE command.)

**Examples** ACQUIRE:STOPAFTER:MODE RUNSTOP sets the instrument to run or stop acquisitions when the user presses the front-panel RUN/STOP button or the user sends the ACQUIRE:STATE command.

ACQUIRE:STOPAFTER:MODE? might return ACQUIRE:STOPAFTER:MODE CONDITION, indicating that the run and stop state of the system is determined by a set of qualifiers specified by the StopAfter condition.

## ALLEV? (Query Only)

This query only command causes the instrument to return all events and their messages and removes the returned events from the Event Queue. The messages are separated by commas. Use the \*ESR? query to enable the events to be returned. For a complete discussion of the use of these registers, see the topics in Status and Events. This command is similar to repeatedly sending \*EVMsg? queries to the instrument.

**Group** Status and Error

**Syntax** ALLEV?

**Related Commands** [\\*CLS](#), [DESE](#), [\\*ESE](#), [\\*ESR?](#), [EVENT?](#), [EVMsg?](#), [EVQty?](#), [\\*SRE](#), [\\*STB?](#)

**Returns** The event code and message in the following format:

```
<Event Code>,<QString>[<Event Code>,<QString>]<QString>::=<Message>;[<Command>]
```

where <Command> is the command that caused the error and may be returned when a command error is detected by the instrument. As much of the command will be returned as possible without exceeding the 60-character limit of the <Message> and <Command> strings combined. The command string is right justified.

**Examples** ALLEV? might return the string :ALLEV 2225,"MEASUREMENT ERROR, NO WAVEFORM TO MEASURE;",420,"QUERY UNTERMINATED;".

## APPLication:ACTivate (No Query Form)

This command (no query form) is used to launch the optional software applications. The application must be installed and activated with a key code. (A key code is not necessary if the application is within a free trial period).

**Group** Miscellaneous

**Syntax** APPLication:ACTivate {"80SJNB" | "80SJARB" | "CRXXX Control" | "IConnect" | "IConnect and MeasureXtractor" | "IConnect and MeasureXtractor Evaluation" | "IConnect S-Parameters and Z-line" | "IConnect S-parameter wizard"}

**Arguments** 80SJNB is the optional 80SJNB Advanced Jitter, Noise, and BER Analysis software for comprehensive jitter, noise, and bit error ratio (BER) analysis for serial data signal impairment characterization.

80SJARB is the optional 80SJARB basic jitter measurement tool, capable of measuring jitter on random or repetitive waveforms.

CRXXX Control is the optional BERTScope Clock Recovery Instrument Control software. This control software works with 80A07, CR125A, CR175A and CR286A clock recovery instruments connected to the DSA8300 via USB.

IConnect is the optional IConnect® software for measurement-based performance evaluation of gigabit interconnect links and devices, including signal integrity analysis, impedance, S-parameter, and eye-diagram tests and fault isolation.

IConnect and MeasureXtractor is the optional software for measurement-based performance evaluation of gigabit interconnect links and devices, including signal integrity analysis, impedance, S-parameter, and eye-diagram tests and fault isolation; includes the ability to extract "black box" interconnect models of passive interconnects.

IConnect and MeasureXtractor Evaluation is trial version of the IConnect and MeasureXtractor software application.

**IConnect S-parameters and Z-line** is the optional software for measurement-based performance evaluation of gigabit interconnect links and devices, including signal integrity analysis, impedance, S-parameters, and fault isolation.

**IConnect S-parameters wizard** is the optional S-parameter “mini-application” that guides you through setup, calibration (including channel deskew) and acquisition of S-parameters, and automatically generates Touchstone files for one or more DUTs.

**Examples**    `APPLICATION:ACTIVATE 80SJNB` launches the 80SJNB software application.

## AUTOSet (No Query Form)

This command (no query form) causes the instrument to adjust its vertical, horizontal, and trigger controls to provide a stable display of the selected waveform. Sending this command is equivalent to pressing the front-panel AUTOSET button.

---

**NOTE.** *If the currently selected waveform is a mask source, the instrument runs a mask autoset.*

*If the TRIGger:SOurce is set to EXTPrescale or C1CLKRec or C3CLKRec and TRIGger:MODe is set to PATtern, the instrument runs a pattern sync autoset.*

*If the currently selected waveform is a channel with a TDR step enabled, the instrument runs a TDR autoset.*

---

**Group**    Miscellaneous

**Syntax**    `AUTOSet EXECute`

**Related Commands**    [AUTOSet:TYPE](#)

**Arguments**    EXECUTE autosets the currently selected waveform.

**Examples**    `AUTOSET EXECUTE` performs an autoset on the currently selected waveform.

## AUTOSet:HORizontal

This command sets or queries whether an autoset should perform the horizontal autosection when AutoSet is executed.

**Group** Horizontal

**Syntax** AUTOSet:HORizontal { ON | OFF | NR1 }  
AUTOSet:HORizontal?

**Related Commands** [AUTOSet](#)

- Arguments**
- ON enables the horizontal AutoSet options.
  - OFF disables the horizontal AutoSet options.
  - NR1 set to 0 disables the horizontal AutoSet options; any other value enables the horizontal AutoSet options.

**Examples** AUTOSET:HORIZONTAL ON enables the horizontal autosection options.  
AUTOSET:HORIZONTAL? might return AUTOSET:HORIZONTAL 0, indicating that horizontal autosection options are disabled.

---

**NOTE.** *A query always returns a 0 or 1.*

---

## AUTOSet:STOP (No Query Form)

This command (no query form) aborts an autosection that was initiated with the AUTOSet command.

**Group** Miscellaneous

**Syntax** AUTOSet:STOP

**Examples** AUTOSET:STOP aborts the autosection.

## AUTOSet:TDR:INCident

This command sets or queries the option to display the TDR incident edge on-screen when performing a TDR Autose. If the Display Incident Edge is on, the incident edge is placed at the one-half horizontal division (from left edge). If the Display Incident Edge is off, the incident edge is placed one horizontal division off the left edge of the screen.

**Group** TDR

**Syntax** AUTOSet:TDR:INCident { ON | OFF | NR1 }  
AUTOSet:TDR:INCident?

**Related Commands** [AUTOSet:TYPE](#), [AUTOSet](#)

**Arguments**

- ON enables the TDR incident edge autose option.
- OFF disables the TDR incident edge autose option.
- NR1 set to 0 disables the TDR incident edge autose option; any other value enables the TDR incident edge autose option.

**Examples** AUTOSET:TDR:INCIDENT ON turns on the Incident Edge option for the TDR autose.

AUTOSET:TDR:INCIDENT? might return :AUTOSET:TDR:INCIDENT 1, indicating that the TDR incident edge is set to be displayed on-screen when performing a TDR autose.

---

**NOTE.** A query always returns a 0 or 1.

---

## AUTOSet:TRIGger

This command sets or queries whether an autose should perform the trigger autose portion when Autose is executed.

**Group** Trigger

**Syntax** AUTOSet:TRIGger { ON | OFF | NR1 }  
AUTOSet:TRIGger?

**Related Commands**    [AUTOSet](#)

- Arguments**
- ON enables the trigger AutoSet options.
  - OFF disables the trigger AutoSet options.
  - NR1 set to 0 disables the trigger AutoSet options; any other value enables the trigger AutoSet options.

**Examples**    AUTOSet:TRIGGER ON enables the trigger autoset options.

AUTOSet:TRIGGER? might return AUTOSet:TRIGGER 0, indicating that trigger autoset options are disabled.

---

**NOTE.** *A query always returns a 0 or 1.*

---

## AUTOSet:TYPE

This command sets or queries the default mode used by the AUTOSet command to perform nonmask and nonTDR AUTOSet operations. Using the TYPE argument, you can bias the autoset operation for display of either a waveform edge, period, eye, or TDR edge in the graticule. Sending this command is equivalent to selecting a mode in the Autoset Properties dialog box.

---

**NOTE.** *When performing MASK:AUTOSet operations, the currently pre-defined mask standard (selected with the MASK:STANDARD command) drives Autoset execution, not the AUTOSet:TYPE command setting. However, when using a user-defined mask, you still might want to use the AUTOSet and AUTOSet:TYPE command, since the MASK:AUTOSet command is unable to obtain autoset parameters from a user-defined mask standard.*

---

**Group**    Miscellaneous

**Syntax**    AUTOSet:TYPE { EDGE | PERIOD | EYE | RZ | TDR }  
 AUTOSet:TYPE?

**Related Commands**    [AUTOSet](#), [AUTOSet:TDR:INCident](#), [AUTOSet:VERTical](#), [AUTOSet:TRIGger](#), [AUTOSet:HORizontal](#)



- Arguments**
- **EDGE** sets subsequent autosegment commands to configure the instrument to display a rising edge of the signal connected to the currently selected channel in the center 20% of the graticule.
  - **PERIOD** sets subsequent autosegment commands to configure the instrument to display 2 to 3 periods of the signal connected to the currently selected channel centered in the graticule.
  - **EYE** sets subsequent autosegment commands to configure the instrument to display one bit (two eye crossings) horizontally centered across approximately 70% of the graticule (7 divisions) and vertically centered across approximately 60% of the graticule (6 divisions). Use the Eye argument for NRZ waveforms.
  - **RZ** sets subsequent autosegment commands to configure the instrument to display one bit (two rising edges) horizontally centered across approximately 60% of the graticule (6 divisions) and vertically centered across approximately 50% of the graticule (5 divisions).
  - **TDR** sets subsequent autosegment commands to configure the instrument to display the first significant reflection at the third horizontal division. If the Display Incident Edge option is on, the incident edge is placed at the one-half horizontal division (from left edge). If the Display Incident Edge is off, the incident edge is placed one horizontal division off the left edge of the screen. If no significant reflections are located, the incident edge is placed at the one-half horizontal division (from left edge) regardless of the Display Incident Edge setting.

---

**NOTE.** If the **Trigger Source** is a clock and the trigger Mode is set to Pattern, an Edge or Period autosegment will attempt to display 20 bit periods.

---

- Examples**
- `AUTOSET:TYPE EDGE` sets the default Autosegment mode to EDGE.
- `AUTOSET:TYPE?` might return `AUTOSET:TYPE RZ`, indicating that the default Autosegment mode is RZ.

## AUTOSet:UNDO (No Query Form)

This command (no query form) returns the parameters changed by an AUTOSet command to their previous states. This command is invalidated the first time you modify any instrument state parameter.

**Group** Miscellaneous

**Syntax** `AUTOSet:UNDO`

**Related Commands**    [AUTOSet](#)

**Examples**    `AUTOSET:UNDO` returns the parameters to their previous states,

## AUTOSet:VERTICAL

This command sets or queries whether an autaset should perform the vertical autaset portion when Autaset is executed.

**Group**    Vertical

**Syntax**    `AUTOSet:VERTical { ON | OFF | NR1 }`  
`AUTOSet:VERTical?`

**Related Commands**    [AUTOSet](#)

- Arguments**
- ON enables the vertical AutoSet options.
  - OFF disables the vertical AutoSet options.
  - NR1 set to 0 disables the vertical AutoSet options; any other value enables the vertical AutoSet options.

**Examples**    `AUTOSET:VERTICAL ON`  
This command enables the vertical autaset options.

`AUTOSET:VERTICAL?` might return `AUTOSET:VERTICAL 0`, indicating that vertical autaset options are disabled.

---

**NOTE.** *A query always returns a 0 or 1.*

---

## BUSY? (Query Only)

This is a query only command that returns the status of the instrument. This command allows you to synchronize the operation of the instrument with your application program. See Synchronization Methods for more detailed information.

**Group**    Status and Error

**Syntax** BUSY?

**Related Commands** \*OPC, \*WAI

**Returns** <NR1>

If 0 is returned, it means that the instrument is not busy processing a command whose execution time is extensive. These commands are listed in the table below.

If 1 is returned, it means that the instrument is busy processing one of the commands listed in the table below.

**Table 2-18: Commands that affect BUSY? response**

Operation	Command
Automatic instrument adjustment	AUTOSet EXECute
Internal self compensation	COMPensate:ALLModules COMPensate:CH<x> COMPensate:DARKLev:CH<x> COMPensate:MAInframe COMPensate:OPTGAIN:CH<x>
Conditional	ACQUIRE:STATE ON or ACQUIRE:STATE RUN (when ACQUIRE:STOPAfter:MODE is set to CONDition)
Hardcopy output	HARDCopy START

**Examples** BUSY? might return 1, indicating that the instrument is busy.

## CALibrate:DATE:CH<x>? (Query Only)

This query only command returns the date and time of the last calibration information update for the sampling module channel.

**Group** Calibration

**Syntax** CALibrate:DATE:CH<x>?

**Examples** CALIBRATE:DATE:CH1? might return CALIBRATE:DATE:CH1 15 JAN 00 16:25, indicating that the last calibration update for the CH1 sampling module was done on January 15, 2000, at 4:25 PM.

## CALibrate:DATE:MAInframe? (Query Only)

This query only command returns the date and time of the last calibration information update for the mainframe.

**Group** Calibration

**Syntax** CALibrate:DATE:MAInframe?

**Examples** CALIBRATE:DATE:MAINFRAME? might return CALIBRATE:DATE:MAINFRAME 15 JAN 00 16:25, indicating that the last calibration update for the mainframe was done on January 15, 2000, at 4:25 PM.

## CALibrate:DCCALibrator

This command sets or queries the value of the DC Calibrator voltage.

**Group** Calibration

**Syntax** CALibrate:DCCALibrator <NR2>  
CALibrate:DCCALibrator?

**Arguments** NR2 is the value to which you want to set the DC Calibrator voltage. The range for this value is from -1.25 VDC through +1.25 VDC.

**Examples** CALIBRATE:DCCALIBRATOR 0.5 sets the DC Calibrator voltage to 0.5 V.  
CALIBRATE:DCCALIBRATOR? might return CALIBRATE:DCCALIBRATOR 5.000000000000E-001, indicating that the DC Calibrator voltage is set to 0.5 V.

## CALibrate:HOSTinfo:CH<x>? (Query Only)

This query only command returns the mainframe channel, model number, and serial number in which the sampling module channel was located during the last calibration information update.

**Group** Calibration

**Syntax** CALibrate:HOSTinfo:CH<x>?

**Examples** CALIBRATE:HOSTINFO:CH2 might return CALIBRATE:HOSTINFO:CH2 "DSA8300,B010235,CH4", indicating that the sampling module currently in channel 2 was installed in channel 4 of the mainframe, serial number B010235, during the last calibration information update.

## CALibrate:LOCK:STATus? (Query Only)

This is a query only command that returns the status of the calibration protection mode.

**Group** Calibration

**Syntax** CALibrate:LOCK:STATus?

**Examples** CALIBRATE:LOCK:STATus? might return CALIBRATE:LOCK:STATUS 1, indicating that the calibration information is protected and cannot be updated.

## CALibrate:STATus:CH<x>? (Query Only)

This is a query only command that returns the current calibration status for the sampling module channel.

**Group** Calibration

**Syntax** CALibrate:STATus:CH<x>?

**Returns** May return PASS, DEFAULTS, or WARMUP.

**Examples** CALIBRATE:STATus:CH1? might return CALIBRATE:STATus:CH1 PASS, indicating that the calibration status of the sampling module in channel 1 is PASS.

## CALibrate:STATus:MAInframe? (Query Only)

This is a query only command that returns the current calibration status for the mainframe.

**Group** Calibration

**Syntax** CALibrate:STATUS:MAInframe?

**Returns** May return PASS, DEFAULTS, or WARMUP.

**Examples** CALIBRATE:STATUS:MAINFRAME? might return  
CALIBRATE:STATUS:MAINFRAME PASS, indicating that the calibration status  
of the mainframe is PASS.

## CALibrate:TEMPerature:CH<x>? (Query Only)

This is a query only command that returns the difference (in °C) between the current sampling module channel temperature and the temperature recorded at the last calibration information update.

**Group** Calibration

**Syntax** CALibrate:TEMPerature:CH<x>?

**Examples** CALIBRATE:TEMPERATURE:CH2? might return  
CALIBRATE:TEMPERATURE:CH2 1.5, indicating that the temperature difference  
between the current sampling module channel temperature and the temperature  
recorded at the last calibration information update is 1.5 °C.

## CALibrate:TEMPerature:MAInframe? (Query Only)

This is a query only command that returns the difference (in °C) between the current mainframe temperature and the temperature recorded at the last calibration information update.

**Group** Calibration

**Syntax** CALibrate:TEMPerature:MAInframe?

**Examples** CALIBRATE:TEMPERATURE:MAINFRAME? might return  
CALIBRATE:TEMPERATURE:MAINFRAME -2.7, indicating that the  
difference in temperature between the current mainframe temperature and the  
temperature recorded at the last calibration information update is -2.7 °C

## CALibrate:UPDATEinfo:ALLModules (No Query Form)

This command (no query form) updates the calibration information for all installed module channels into nonvolatile memories.

---

**NOTE.** *This command is for service personnel only. Refer to the calibration procedures in the Service Manual.*

---

**Group** Calibration

**Syntax** CALibrate:UPDATEinfo:ALLModules

**Examples** CALIBRATE:UPDATEINFO:ALLMODULES updates the nonvolatile memories of all installed module channels with the latest calibration information.

## CALibrate:UPDATEinfo:CH<x> (No Query Form)

This command (no query form) updates the calibration information in the nonvolatile memory of the sampling module. This includes the current date and time, temperature, and host information (such as current channel, location, model number, and serial number).

---

**NOTE.** *This command is for service personnel only. Refer to the calibration procedures in the Service Manual.*

---

**Group** Calibration

**Syntax** CALibrate:UPDATEinfo:CH<x>

**Examples** CALIBRATE:UPDATEINFO:CH1 updates the nonvolatile memory of the module occupying channel 1 with the latest calibration information.

## CALibrate:UPDATEinfo:MAInframe (No Query Form)

This command (no query form) updates the calibration information in the nonvolatile memory of the mainframe. This includes the current date, time, and temperature.

---

**NOTE.** *This command is for service personnel only. Refer to the calibration procedures in the Service Manual.*

---

**Group** Calibration

**Syntax** CALibrate:UPDATEinfo:MAInframe

**Examples** CALIBRATE:UPDATEINFO:MAINFRAME updates the nonvolatile memory of the mainframe with the latest calibration information.

## CH<x>? (Query Only)

This is a query only command that returns the vertical parameters for the specified channel. The channel is specified by x, which can be 1 through 8.

**Group** Vertical

**Syntax** CH<x>?

**Examples** CH3? might return CH3:DESKEW 0.000000000000; OFFSET 0.000000000000; UNITS AUTO; FILTER:VALUE NONE; CH3:EXTATTEN:MODE LINEAR; VALUE 1.000000000000; CH3:POSITION 0.000000000000; PTPEAK 0.700000000000; SCALE 0.700000000000.

## CH<x>:BANdwidth:LIST? (Query Only)

This query-only command returns a list of the available bandwidths for the specified channel. The channel is specified by x, which can be 1 through 8. The returned list contains the actual bandwidths available for the channel. The available bandwidths are dependent on the module.

The optical modules have differing valid settings. Use the <>:LIST? query to determine the valid {Clock Recovery | Filter | Bandwidth | Wavelength} settings for your optical module. Features of currently installed optical modules can be viewed using the System Properties dialog.

To set the bandwidth for a channel, use the CH<x>:BANdwidth:VALue command.

**Group** Vertical



**Syntax** CH<x>:BANDwidth:LIST?

**Related Commands** CH<x>:BANDwidth:VALue

**Examples** CH1:BANDWIDTH:LIST? might return CH1:BANDWIDTH:LIST  
20.000000000E+009,12.49999974400E+009, indicating the bandwidths  
available for Channel 1.

## CH<x>:BANDwidth:VALue

This command sets or queries the bandwidth for the channel specified by x, which can be 1 through 8. Sending the command is the equivalent to selecting a bandwidth from the Bandwidth pulldown menu in the Signal Conditioning section of the Vertical Setup dialog box for sampling modules. The sampling module determines the available bandwidths.

You can use the CH<x>:BANDwidth:LIST? query to determine the available bandwidths for the specified channel.

**Group** Vertical

**Syntax** CH<x>:BANDwidth:VALue <NR3>  
CH<x>:BANDwidth:VALue?

**Arguments** NR3 is the value of the bandwidth you want to set. Use the CH<x>:BANDwidth:LIST? command to query the bandwidths available for the specified channel. The instrument automatically selects the bandwidth nearest to the input value.

**Examples** CH1:BANDWIDTH:VALUE 2.0E+10 sets the Channel 1 bandwidth to 20 GHz.  
CH2:BANDWIDTH:VALUE? might return CH2:BANDWIDTH:VALUE  
12.49999974400E+009, indicating that the bandwidth for Channel 2 is set  
to 12.5 GHz.

## CH<x>:DELay

This command sets or queries the delay time for the channel specified by x, which can be 1 through 8. Sending this command is the equivalent to setting Delay in the Vertical Setup dialog box for the selected channel. You can adjust delay to add or subtract an independent, channel-based acquisition time delay to the delay from the trigger point (the horizontal position) that is inherently common to the

acquisition of all channels. This allows you to compensate for delay differences introduced between channels by external cabling with unequal delays.

---

**NOTE.** *Delay is only supported by sampling modules that include delay hardware. Using delay instead of deskew will not affect overall acquisition performance since multiple passes for acquisitions are not required.*

---

**Group** Vertical

**Syntax** CH<x>:DElay<NR3>  
CH<x>:DElay?

**Arguments** NR3 is the delay time for this channel. The range is -100% to +100% with a resolution of 0.1%.

**Examples** CH7:DELAY 0.1 sets the delay for Channel 7 to 10%.  
CH3:DELAY? might return CH3:DELAY 0.250000000 indicating that the delay time for Channel 3 is set to 25%.

## CH<x>:DESkew

This command sets or queries the deskew time for the channel specified by x, which can be 1 through 8. Sending this command is the equivalent to setting Deskew in the Vertical Setup dialog box for the selected channel.

You can adjust deskew to add or subtract an independent, channel-based acquisition time delay to the delay from the trigger point (the horizontal position) that is inherently common to the acquisition of all channels. This allows you to compensate for delay differences introduced between channels by external cabling with unequal delays.

---

**NOTE.** *Setting different deskew values on dual-channel sampling modules that use a single sampling strobe (for example, the 80E02, 80E03, 80E04 sampling modules) reduces overall acquisition performance. In these instances, the acquisition must digitize each channel on a separate acquisition pass to obtain each of the signals with the desired time alignment. This multiple pass acquisition process applies to the acquisition of the channels within each active timebase.*

---

**Group** Vertical

<b>Syntax</b>	CH<x>:DESkeW <NR3> CH<x>:DESkeW?
<b>Arguments</b>	NR3 is the deskew time for this channel. The range is -0.5 ns to +100 ns with a resolution of 1 ps. Out of range values are clipped.
<b>Examples</b>	CH7:DESKEW 5.0E-9 sets the deskew time for Channel 7 to 5 ns. CH3:DESKEW? might return CH3:DESKEW 10.0000000008E-009, indicating that the deskew time for Channel 3 is set to 10 ns.

## CH<x>:ERCAL? (Query Only)

This query-only command returns whether calibrated extinction ratio is available for the optical module for the specified channel. The channel is specified by x, which can be 1 through 8.

<b>Group</b>	Vertical
<b>Syntax</b>	CH<x>:ERCAL?
<b>Examples</b>	CH1:ERCAL? might return :CH1:ERCAL 0, indicating that the calibrated extinction ratio is not available for the optical module in the channel 1.

---

**NOTE.** A query always returns a 0 or 1.

---

## CH<x>:EXTAtten:MODE

This command sets or queries the external attenuation mode for the channel specified by x, which can be 1 through 8.

<b>Group</b>	Vertical
<b>Syntax</b>	CH<x>:EXTAtten:MODE { LINEar   DB } CH<x>:EXTAtten:MODE?
<b>Arguments</b>	<ul style="list-style-type: none"> <li>■ LINEar sets the mode so that the attenuation value is a linear numeric multiplier, such as 15X.</li> <li>■ DB sets the mode so that the attenuation value is in dB.</li> </ul>

**Examples** CH7:EXTATTEN:MODE LIN sets the external attenuation mode to linear for the Channel 7.

CH3:EXTATTEN:MODE? might return CH3:EXTATTEN:MODE DB, indicating that the external attenuation is set to the DB mode for Channel 3.

## CH<x>:EXTAtten:VALue

This command sets a value matching the amount of attenuation applied externally, before the signal enters the specified input channel. The channel is specified by *x*, which can be 1 through 8. The query form returns the amount of attenuation that is set for the instrument. The instrument takes into account this external attenuation when displaying vertical readouts, measurements, and so on for this channel. This command is the equivalent to entering a value in the External Attenuation control in the External Attenuation section of the Vertical Setup dialog box.

Use the CH<x>:EXTAtten:MODE command to select whether the attenuation value is set or returned as a linear attenuation (such as 15X) or as dB attenuation (such as 10 dB).

---

**NOTE.** *External Attenuation (dB) = 20 \* log10 (Linear) for External Attenuation (linear) > 0.*

---

For example, 0 dB=1X, 20 dB=10X, 40 dB=100X, and so on.

The external attenuation is set to 1 (linear) or 0 dB for all TDR sources and for all channels released as TDR sources. External attenuation is set to 1 (linear) or 0 dB when you attach an external real-time accessory, such as a probe, to the selected channel.

**Group** Vertical

**Syntax** CH<x>:EXTAtten:VALue <NR3>  
CH<x>:EXTAtten:VALue?

**Related Commands** [CH<x>:EXTAtten:MODE](#)

**Arguments** NR3 is the attenuation value specified as a multiplier in the range of 1.0E-6 to 1.0E+6. The default is 1.00.

**Examples** CH7:EXTATTEN:VALUE 1.5E+1 sets the external attenuation for Channel 7 to 15X when linear is set as the external attenuation mode.

CH3:EXTATTEN:VALUE? might return CH3:EXTATTEN:VALUE  
10.000000000000, indicating that the external attenuation is set to 10X for  
Channel 3 when linear is set as the external attenuation mode.

## CH<x>:FILTer:LIST? (Query Only)

This query only command returns a list of the filters available for the channel specified by x, which can be 1 through 8. The filters available depend on the module. Sending this command is the equivalent using the Filter pulldown in the Signal Conditioning section of the Optical portion of the Vertical Setup dialog box.

Use the CH<x>:FILTer:VALue command to set a filter.

**Group** Vertical

**Syntax** CH<x>:FILTer:LIST?

**Related Commands** [CH<x>:FILTer:VALue](#)

**Examples** CH1:FILTer:LIST? might return CH1:FILTer:LIST  
NONE,OC12,OC48,OC192, indicating the filters available for Channel  
1.

## CH<x>:FILTer:VALue

This command sets or queries the reference-receiver filter that is applied to the channel specified by x, which can be 1 through 8. The reference-receiver filter selections require hardware support that may not be provided by your sampling module; see your module user manual to determine which filters your module supports. You can also use the CH<x>:FILTer:LIST? query to return a list of the filters available for the specified channel.

If you select a filter that your sampling module does not support, you will receive Execution Error code 2508, with error message "Selected filter is not available". See Messages for general information on programming interface messages.

**Group** Vertical

**Syntax** CH<x>:FILTer:VALue { NONE | CPRI7373 | ENET103R4 |  
ENET103R10 | ENET257R4 | ENET1250 | ENET2500 | ENET3125 |  
ENET41250 | ENET9953 | ENET10313 | ENET11096 | FC1063 |  
FC2125 | FC3188 | FC4250 | FC8500 | FC8500FINAL | FC10519 |

FC11317 | FC14025 | FEC2666 | FEC42657 | FEC43018 | FEC10664  
 | FEC10709 | FEC12500 | FEC43018 | FPELECOUT | INF2500 |  
 INF5000 | INF14063 | INF25781 | INFIniband | PCIX2500 |  
 PCIX6250 | OBSAI6144 | OC1 | OC3 | OC9 | OC12 | OC18 | OC24  
 | OC36 | OC48 | OC96 | OC192 | OC768 | OTU27952 | SATA1500 |  
 SATA3000 | SATA6000 | SFEC12500 | VSR5-3318 }

CH<x>:FILTER:VALue?

**Related Commands**    [CH<x>:FILTER:LIST?](#)

**Arguments**

NONE disables reference-receiver filtering.

CPRI7373 selects a reference-receiver filter for the 7.373 Gb/s bit rate, in accordance with the Wireless Base Station CPRI 7.373 standard.

ENET103R4 selects a reference-receiver filter for the 10.3125 Gb/s bit rate, in accordance with the Ethernet 40GBASE-LR4/SR4 standard.

ENET103R10 selects a reference-receiver filter for the 10.3125 Gb/s bit rate, in accordance with the Ethernet 100GBASE-SR10 standard.

ENET257R4 selects a reference-receiver filter for the 25.78125 Gb/s bit rate, in accordance with the Ethernet 100GBASE-ER4/LR4 standard.

ENET1250 selects a reference-receiver filter for the 1.250 Gb/s bit rate, in accordance with the IEEE 802.3ae-2002 standard.

ENET2500 selects a reference-receiver filter for the 2.500 Gb/s bit rate (2x Gigabit Ethernet).

ENET3125 selects a reference-receiver filter for the 3.125 Gb/s bit rate, in accordance with the IEEE 802.3ae-2002 standard.

ENET9953 selects a reference-receiver filter for the 9.95328 Gb/s bit rate, in accordance with the IEEE 802.3ae-2002 standard.

ENET10313 selects a reference-receiver filter for the 10.3125 Gb/s bit rate, in accordance with the IEEE 802.3ae-2002 standard.

ENET11096 selects a reference-receiver filter for the 11.095728 Gb/s bit rate, which is a forward error-correction rate for 10GbE (10.3125 Gb/s per IEEE 802.3ae-2002 standard).

ENET41250 selects a reference-receiver filter for the 41.250 Gb/s bit rate, in accordance with the Ethernet 40GBASE-FR standard.

FC1063 selects a reference-receiver filter for the 1.0625 Gb/s bit rate, in accordance with the ANSI X3.230-1994 standard.

FC2125 selects a reference-receiver filter for the 2.125 Gb/s bit rate, in accordance with the ANSI Fibre Channel Physical Interface (FC-PI) Rev 11 draft standard.

**FC3188** selects clock recovery for the 3.1875 Gb/s bit rate, in accordance with the proposed 10 Gigabit Fibre Channel (10GFC) standard.

**FC4250** selects a reference-receiver filter for the 4.250 Gb/s bit rate, in accordance with the ANSI Fibre Channel Physical Interface (FC-PI) Rev 11 draft standard.

**FC8500** selects a reference-receiver filter for the 8.5000 Gb/s bit rate, in accordance with the preliminary 8GFC standard.

**FC8500FINAL** selects a reference-receiver filter for the 8.5000 Gb/s bit rate, in accordance with the Fibre Channel Optical 8GFC standard.

**FC10519** selects a reference-receiver filter for the 10.518750 Gb/s bit rate, in accordance with the proposed 10 Gigabit Fibre Channel (10GFC) standard.

**FC11317** selects a reference-receiver filter for the 11.317 Gb/s bit rate, in accordance with the proposed 10 Gigabit Fibre Channel (10GFC) standard.

**FC14025** selects a reference-receiver filter for the 14.02500 Gb/s bit rate, in accordance with the Fibre Channel Optical 16GFC MM/SM r6.1 standard.

**FEC2666** selects a reference-receiver filter for the 2.6660571 Gb/s bit rate, in accordance with the ITU-T G.975 Recommendation.

**FEC10664** selects a reference-receiver filter for the 10.66423 Gb/s bit rate, in accordance with the ITU-T G.975 Recommendation.

**FEC10709** selects a reference-receiver filter for the 10.709225 Gb/s bit rate, in accordance with the ITU-T G.709 draft Recommendation.

**FEC12500** selects a reference-receiver filter for the 12.5000 Gb/s bit rate, in accordance with the Fibre Channel Optical 12.5G FEC standard. Replaces the **SFEC12500** argument. If **SFEC12500** is set and the filter is queried, the instrument returns **FEC12500**.

**FEC42657** selects a reference-receiver filter for the 42.657 Gb/s bit rate, in accordance with the ITU-T G.709 Recommendation.

**FEC43018** selects a reference-receiver filter for the 43.018414 Gb/s bit rate, in accordance with the ITU-T G.709 Draft Recommendation.

**FPELECOUT** enables the full power electrical output for optical modules. This is not a filter.

**INF2500** selects a reference-receiver filter for the 2.50 Gb/s bit rate, in accordance with the Infiniband Optical standard. Replaces the **INFINIBAND** argument. If **INFINIBAND** is set and the filter is queried, the instrument returns **INF2500**.

**INF5000** selects a reference-receiver filter for the 5.0 Gb/s bit rate, in accordance with the Infiniband Optical standard.

**INF14063** selects a reference-receiver filter for the 14.06250 Gb/s bit rate, in accordance with the Infiniband FDR standard.

INF25781 selects a reference-receiver filter for the 25.781 Gb/s bit rate, in accordance with the Infiniband EDR standard.

OBSAI6144 selects a reference-receiver filter for the 6.144 Gb/s bit rate, in accordance with the Wireless Base Station OBSAI standard.

OC1 selects a reference-receiver filter for the 51.84 Mb/s bit rate, in accordance with the Sonet/SDH OC-1/STM-0 standard.

OC3 selects a reference-receiver filter for the 155.52 Mb/s bit rate, in accordance with the Sonet/SDH OC-3/STM-1 standard.

OC9 selects a reference-receiver filter for the 466.56 Mb/s bit rate, in accordance with the Sonet/SDH OC-9/STM-3 standard.

OC12 selects a reference-receiver filter for the 622.08 Mb/s bit rate, in accordance with the Sonet/SDH OC-12/STM-4 standard.

OC18 selects a reference-receiver filter for the 933.12 Mb/s bit rate, in accordance with the Sonet/SDH OC-18/STM-6 standard.

OC24 selects a reference-receiver filter for the 1.24416 Gb/s bit rate, in accordance with the Sonet/SDH OC-24/STM-8 standard.

OC36 selects a reference-receiver filter for the 1.86624 Gb/s bit rate, in accordance with the Sonet/SDH OC-36/STM-12 standard.

OC48 selects a reference-receiver filter for the 2.48832 Gb/s bit rate, in accordance with the Sonet/SDH OC-48/STM-16 standard.

OC96 selects a reference-receiver filter for the 4.97664 Gb/s bit rate, in accordance with the Sonet/SDH OC-96/STM-32 standard.

OC192 selects a reference-receiver filter for the 9.95328 Gb/s bit rate, in accordance with the Sonet/SDH OC-192/STM-64 standard.

OC768 selects a reference-receiver filter for the 39.8131 Gb/s bit rate, in accordance with the Sonet/SDH OC-768/STM-256 standard.

OTU27952 selects a reference-receiver filter for the 27.95 Gb/s bit rate, in accordance with the Fibre Channel Optical OTU4 standard.

### Examples

CH1:FILTER:VALUE NONE disables the filtering for Channel 1.

CH2:FILTER:VALUE? might return CH2:FILTER:VALUE OC12, indicating that the Sonet/SDH OC-12/STM-4 filter is enabled for Channel 2.

## CH<x>:OFFSet

This command sets or queries the vertical offset for the specified channel. The channel is specified by x, which can be 1 through 8. Setting an offset with this command is the equivalent to entering a value in the Offset control in the Channel



section of the Vertical Setup dialog box or adjusting the front-panel Vertical OFFSET knob.

Channel offset adjusts the vertical acquisition window (moves the DC level around which the signal is sampled) for the selected channel. Visualize offset as scrolling the acquisition window towards the top of a large signal (or one with a positive DC bias) for increased offset values and scrolling towards the bottom for decreased offset values. Optimal vertical accuracy (of the entire signal) is obtained when the offset is set to the average DC bias of the input signal (that is, when it centers the signal around the input dynamic range of the sampling module).

The range and resolution of offset values for a given channel is dependent on multiple factors: the sampling module type (specifically, its input dynamic range), the probe type (if attached), the external attenuation factor, the selected units (if TDR is active), and the scale.

For more information on how offset ranges are determined, refer to the topic Vertical Offset in the online help for your instrument.

**Group** Vertical

**Syntax** CH<x>:OFFSet <NR3>  
CH<x>:OFFSet?

**Related Commands** [CH<x>:POSition](#)

**Arguments** NR3 is the offset value for the specified channel; the range is  $\pm 1.6$  V.

**Examples** CH4:OFFSET 2.0E-3 sets the offset for Channel 4 to 2 mV.

CH3:OFFSET? might return CH3:OFFSET 0.100000000000, indicating that the offset for Channel 3 is set to 100 mV.

## CH<x>:POSition

This command sets or queries the vertical position of the specified channel. The channel is specified by x, which can be 1 through 8. Sending this command is the equivalent to setting Position in the Setup section of the Vertical Setup dialog box or adjusting the front-panel Vertical Position knob.

Increasing the position value of a waveform causes the waveform to move up, and decreasing the position value causes the waveform to move down. Position adjusts only the display position of a channel, math, or reference waveform.

<b>Group</b>	Vertical
<b>Syntax</b>	CH<x>:POSITION <NR3> CH<x>:POSITION?
<b>Related Commands</b>	<a href="#">CH&lt;x&gt;:OFFSet</a>
<b>Arguments</b>	<NR3> is the position value in divisions from the center graticule. The range is $\pm 5$ divisions for channels and $\pm 1000$ divisions for math and reference waveforms.
<b>Examples</b>	CH2:POSITION 1.3E+00 positions the Channel 2 input signal 1.3 divisions above the center graticule.  CH1:POSITION? might return CH1:POSITION -2.000000000000, indicating that the current position of Channel 1 is 2 divisions below the center graticule.

## CH<x>:SCALE

This command sets or queries the vertical scale of the channel specified by x, which can be 1 through 8. Sending this command is the equivalent to setting a value in the Scale control in the Setup section of the Vertical Setup dialog box or adjusting the front-panel Vertical SCALE knob.

Channel scale, expressed as the per division setting, adjusts the display size for the selected channel. Similar to the scale control for reference and math waveforms, this control provides graphical scaling only; the control has no effect on the input vertical acquisition hardware. Increasing the scale causes the waveform to be displayed smaller. Decreasing the scale causes the waveform to be displayed larger.

The range and resolution of scale values for a given channel is dependent on multiple factors: the sampling module type (specifically, its input dynamic range), the probe type (if attached), the external attenuation factor, and the selected units (if TDR is active).

For more information on how scale ranges are determined, refer to the topic Vertical Scale in the online help for your instrument.

Channel scale changes have an effect on the allowable offset range for the channel. In general, as the vertical scale value is reduced, the amount of allowable offset increases.

For more information on how scale affects offset range, refer to the topic Vertical Offset in the online help for your instrument.

<b>Group</b>	Vertical
<b>Syntax</b>	CH<x>:SCALE <NR3> CH<x>:SCALE?
<b>Related Commands</b>	<a href="#">CH&lt;x&gt;:OFFSet</a> , <a href="#">CH&lt;x&gt;:POSition</a>
<b>Arguments</b>	NR3 is the vertical channel scale in units per division.
<b>Examples</b>	CH4:SCALE 5.0E-01 sets the scale for Channel 4 to 500 mV per division. CH2:SCALE? might return CH2:SCALE 0.02000000000, indicating that the current scale setting of Channel 2 is 20 mV per division.

## CH<x>:UNIts

This command sets or queries the units for the specified channel. The channel is specified by x, which can be 1 through 8. Sending this command is the equivalent to setting the Units control in the Channel section of the Vertical Setup dialog box. You can use this control to override the automatic selection of units for the selected channel. If you set this value to AUto, the instrument automatically selects the units. If you set this value to Volt, Watt, or Amp, the instrument will apply these units to the vertical axis.

- The selections in this control are Auto, Volt, Amp, and Watt. When you change the units setting, all vertical units are reported as specified and no data transformations take place.
- For TDR waveforms, the Units control in the Vertical Setup dialog box is not active. Use the TDR:CH<x>:UNIts command to set and query units in TDR. The units available are Rho, Ohm, and Volt. When you change units of TDR waveforms, data transformations take place. The default is Rho.

<b>Group</b>	Vertical
<b>Syntax</b>	CH<x>:UNIts { Auto   Volt   Watt   Amp } CH<x>:UNIts?

- Arguments**
- `AUTO` sets the units for the specified channel so that they are automatically determined by the instrument (depends on the probe attached to the specified channel).
  - `volt` sets the vertical-axis units for the specified channel to volts.
  - `watt` sets the vertical-axis units for the specified channel to watts.
  - `Amp` sets the vertical-axis units for the specified channel to amperes.

**Examples**

`CH4:UNITS AUTO` sets the vertical axis units for Channel 4 so that they are determined automatically by the instrument.

`CH1:UNITS?` might return `CH1:UNITS VOLT`, indicating that the vertical-axis units for Channel 1 are set to volts.

## CH<x>:WFMLabel

This command sets or queries the label associated with the channel waveform specified.

**Group** Vertical

**Syntax** `CH<x>:WFMLabel <Qstring>`  
`CH<x>:WFMLabel?`

**Related Commands** [MATH<x>:WFMLabel](#), [REF<x>:WFMLabel](#)

**Arguments** `Qstring` sets the label for the waveform.

**Examples**

`CH1:WFMLABEL "MY CH1 WAVEFORM"` sets the label for CH1 to be "My CH1 waveform," which is the label displayed with the waveform when it is displayed on screen.

`CH1:WFMLABEL?` might return `CH1:WFMLABEL "MY CH1 WAVEFORM"`, indicating that the label for channel 1 is set to "My CH1 waveform".

## CH<x>:WLENgth:LIST? (Query Only)

This query only command returns a list of the available wavelengths for the channel specified by x, which can be 1 through 8. Generally, only optical modules in slots CH1 or CH3 have wavelength selections. The returned list contains the

actual wavelengths available for the channel. Use the `CH<x>:WLENgth:VALue` command to set a value from the list.

**Group** Vertical

**Syntax** `CH<x>:WLENgth:LIST?`

**Related Commands** [CH<x>:WLENgth:VALue](#)

**Examples** `CH2:WLENGTH:LIST?` might return `CH2:WLENLIST FACTORY,1550,1310,USER,1330`, indicating the available wavelengths for Channel 2.

## CH<x>:WLENgth:VALue

This command sets or queries the wavelength for the channel specified by `x`, which can be 1 through 8. Generally, only optical modules in slots CH1 or CH3 have filter selections. You can use the `CH<x>:WLENgth:LIST?` query to determine the available wavelengths.

**Group** Vertical

**Syntax** `CH<x>:WLENgth:VALue <NR3> [ USER | FACTory ]`  
`CH<x>:WLENgth:VALue?`

**Related Commands** [CH<x>:WLENgth:LIST?](#)

- Arguments**
- NR3 specifies the wavelength.
  - USER must follow a user-calibrated value.
  - FACTory must follow a factory-calibrated value.

**Examples** `CH1:WLENGTH:VALUE 1310` sets the Channel 1 wavelength to 1310.  
`CH2:WLENGTH:VALUE?` might return `CH2:WLENGTH:VALUE 1550`, indicating that the wavelength for Channel 2 is set to 1550.

## \*CLS (No Query Form)

This command (no query form) clears the following status data structures of the instrument:

- Event Queue
- Standard Event Status Register (SESR)
- 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. (For more information about these registers and bits, and of event handling in general, see the Status and Events Overview and its subtopics.)

\*CLS can suppress a service request that is to be generated by an \*OPC. This will happen if a hardcopy output or conditional acquisition operation is still being processed when the \*CLS command is executed.

**Group** Status and Error

**Syntax** \*CLS

**Related Commands** DESE, \*ESE, \*ESR?, EVENT?, EVMsg?, \*SRE, \*STB?

**Examples** \*CLS clears the instrument status data structures.

## COMPensate:ALLModules (No Query Form)

This command (no query form) compensates all installed module channels for DC variances.




---

**CAUTION.** *Before sending this command, disconnect or disable all signals and probes from all sampling input channels. Ensure that all electrical inputs are terminated to 50  $\Omega$  and all optical module inputs have dust covers installed before continuing. For best results, ensure that the mainframe compensation is valid and leave trigger sources for the selected channels connected and active.*

---



---

**NOTE.** *Data from a successful compensation is not automatically saved into nonvolatile user storage. Use the COMPensate:SAVe commands if you want to retain the data in nonvolatile storage.*

---

---

<b>Group</b>	Compensation
<b>Syntax</b>	COMPensate:ALLModules
<b>Related Commands</b>	<a href="#">COMPensate:SAVe:USER:ALLModules</a>
<b>Examples</b>	COMPENSATE:ALLMODULES performs the compensation routines on all installed module channels.

## COMPensate:CH<x> (No Query Form)

This command (no query form) compensates the module channel for DC variances.



---

**CAUTION.** Before sending this command, disconnect or disable all signals and probes from all sampling input channels. Ensure that all electrical inputs are terminated to 50  $\Omega$  and all optical module inputs have dust covers installed before continuing. For best results, ensure that the mainframe compensation is valid and leave trigger sources for the selected channel connected and active.

---

---

**NOTE.** Data from a successful compensation is not automatically saved into nonvolatile user storage. Use the COMPensate:SAVe commands if you want to retain the data in nonvolatile storage.

---

<b>Group</b>	Compensation
<b>Syntax</b>	COMPensate:CH<x>
<b>Related Commands</b>	<a href="#">COMPensate:SAVe:USER:CH&lt;x&gt;</a>
<b>Examples</b>	COMPENSATE:CH1 performs the compensation routines for module occupying channel 1.

## COMPensate:DARKLev:CH<x> (No Query Form)

This command (no query form) compensates an optical channel by removing residual DC offsets in the entire vertical path. This provides better DC offset compensation at the current vertical settings.



**CAUTION.** *Before sending this command, disconnect or disable all signals and probes from all sampling input channels. Ensure that all electrical inputs are terminated to 50  $\Omega$  and all optical module inputs have optical signals disabled or dust covers installed before continuing. For best results, ensure that the mainframe compensation is valid and leave trigger sources for the selected channel connected and active.*

---

**Group** Compensation

**Syntax** COMPensate:DARKLev:CH<x>

**Examples** COMPENSATE:DARKLEV:CH1 performs the darklevel compensation on the optical module occupying channel 1.

### COMPensate:DATE:CH<x>? (Query Only)

This is a query only command that returns the date and the time of the current in-use (that is, run-time) compensation data for the module channel.

**Group** Compensation

**Syntax** COMPensate:DATE:CH<x>?

**Examples** COMPENSATE:DATE:CH2? might return COMPENSATE:DATE:CH2 15 JAN 00 16:25

### COMPensate:DATE:MAInframe? (Query Only)

This is a query only command that returns the date and the time of the current in-use (that is, run-time) compensation data for the mainframe.

**Group** Compensation

**Syntax** COMPensate:DATE:MAInframe?

**Examples** COMPENSATE:DATE:MAINFRAME? might return  
COMPENSATE:DATE:MAINFRAME 15 JAN 00 16:25



## COMPensate:MAInframe (No Query Form)

This command (no query form) compensates the mainframe for DC variances.



**CAUTION.** Before sending this command, disconnect or disable all signals and probes from all sampling input channels and mainframe trigger inputs. Ensure that all electrical inputs are terminated to 50  $\Omega$  and all optical module inputs have dust covers installed before continuing.

**NOTE.** Data from a successful compensation is not automatically saved into nonvolatile user storage. Use the COMPensate:SAVe commands if you want to retain the data in nonvolatile storage.

**Group** Compensation

**Syntax** COMPensate:MAInframe

**Related Commands** [COMPensate:SAVe:USER:MAInframe](#)

**Examples** COMPENSATE:MAINFRAME performs the compensation routine on the mainframe.

## COMPensate:OPTGAIN:CH<x> (No Query Form)

This command (no query form) compensates the gain of the optical channel by comparing the given input power to an average optical power measurement. The gain correction factor and the given input wavelength are automatically stored in the nonvolatile memory of the sampling module channel. The optical signal source must be connected and configured appropriately before sending this command. You should use a signal of known optical power at the wavelength for which you are compensating.

**Group** Compensation

**Syntax** COMPensate:OPTGAIN:CH<x> <NR1>, <NR3>

- Arguments**
- NR1 argument is the wavelength (in nm) of the input optical signal.
  - NR3 argument is the average optical power of the input optical signal.

**Examples**    `COMPENSATE:OPTGAIN:CH1 1000,1.0E-3`

## **COMPensate:RECALL:FACTory:ALLModules (No Query Form)**

This command (no query form) recalls compensation data for all installed module channels from their respective nonvolatile factory memories into the current, in-use (that is, run-time) compensation memory.

**Group**    Compensation

**Syntax**    `COMPensate:RECALL:FACTory:ALLModules`

**Examples**    `COMPENSATE:RECALL:FACTORY:ALLMODULES`

## **COMPensate:RECALL:FACTory:CH<x> (No Query Form)**

This command (no query form) recalls compensation data from the nonvolatile factory memory of the module channel into its associated in-use (that is, run-time) compensation memory.

**Group**    Compensation

**Syntax**    `COMPensate:RECALL:FACTory:CH<x>`

**Examples**    `COMPENSATE:RECALL:FACTORY:CH2`

## **COMPensate:RECALL:FACTory:MAInframe (No Query Form)**

This command (no query form) recalls compensation data from the nonvolatile factory memory of the mainframe into its associated in-use (that is, run-time) compensation memory.

**Group**    Compensation

**Syntax**    `COMPensate:RECALL:FACTory:MAInframe`

**Examples**    `COMPENSATE:RECALL:FACTORY:MAINFRAME`

## COMPensate:RECALL:USER:ALLModules (No Query Form)

This command (no query form) recalls compensation data for all installed module channels from their respective nonvolatile user memories into the current, in-use (that is, run-time) compensation memory.

**Group** Compensation

**Syntax** COMPensate:RECALL:USER:ALLModules

**Examples** COMPENSATE:RECALL:USER:ALLMODULES

## COMPensate:RECALL:USER:CH<x> (No Query Form)

This command (no query form) recalls compensation data from the nonvolatile user memory of the module channel into its associated in-use (that is, run-time) compensation memory.

**Group** Compensation

**Syntax** COMPensate:RECALL:USER:CH<x>

**Examples** COMPENSATE:RECALL:USER:CH1

## COMPensate:RECALL:USER:MAInframe (No Query Form)

This command (no query form) recalls compensation data from the nonvolatile user memory of the mainframe into its associated in-use (that is, run-time) compensation memory.

**Group** Compensation

**Syntax** COMPensate:RECALL:USER:MAInframe

**Examples** COMPENSATE:RECALL:USER:MAINFRAME

## COMPensate:RESults? (Query Only)

This is a query only command that returns an abbreviated status about the results of the last compensation execution. Any result other than PASS generally indicates a failure. For a more detailed message about the results of the last compensation execution, use the COMPensate:RESults:VERBose? query.

<b>Group</b>	Compensation
<b>Syntax</b>	COMPensate:RESults?
<b>Examples</b>	COMPENSATE:RESULTS? might return COMPENSATE:RESULTS Pass, indicating that the compensation was successful.

## COMPensate:RESults:VERBose? (Query Only)

This is a query only command that returns a more explanatory message than the COMPensate:ReSults query returns about the results of the last compensation execution.

<b>Group</b>	Compensation
<b>Syntax</b>	COMPensate:RESults:VERBose?
<b>Related Commands</b>	<a href="#">COMPensate:RESults?</a>
<b>Examples</b>	COMPENSATE:RESULTS:VERBOSE? might return COMPENSATE:RESULTS:VERBOSE COMP COMPLETED, NO FAILURES.

## COMPensate:SAVe:USER:ALLModules (No Query Form)

This command (no query form) saves the volatile run-time compensation data for all installed module channels into their respective nonvolatile user memories.

<b>Group</b>	Compensation
<b>Syntax</b>	COMPensate:SAVe:USER:ALLModules

**Examples**    `COMPENSATE:SAVE:USER:ALLMODULES`

## **COMPensate:SAVe:USER:CH<x> (No Query Form)**

This command (no query form) saves the volatile run-time compensation data for the module channel into the nonvolatile user memory of the module channel. The channel is specified by x, which can be 1 through 8.

**Group**    Compensation

**Syntax**    `COMPensate:SAVe:USER:CH<x>`

**Examples**    `COMPENSATE:SAVE:USER:CH2`

## **COMPensate:SAVe:USER:MAInframe (No Query Form)**

This command (no query form) saves the volatile run-time compensation data for the mainframe into the nonvolatile user memory of the mainframe.

**Group**    Compensation

**Syntax**    `COMPensate:SAVe:USER:MAInframe`

**Examples**    `COMPENSATE:SAVE:USER:MAINFRAME`

## **COMPensate:STATus:CH<x>? (Query Only)**

This is a query only command that returns the current compensation status for the module channel. Possible responses are DEFaults, WARMup, FAIL, PASS, and COMPReq.

**Group**    Compensation

**Syntax**    `COMPensate:STATus:CH<x>?`

**Examples**    `COMPENSATE:STATUS:CH1?` might return `COMPENSATE:STATUS:CH1`  
`COMPREQ`, indicating that the warm-up period for the instrument has elapsed, but

the current compensation temperature delta is greater than desired, or that the specified module has been moved to a different module compartment since last compensated. In either case, the instrument should be compensated again.

## COMPensate:STATus:MAInframe? (Query Only)

This is a query only command that returns the current compensation status for the mainframe. Possible responses are DEFaults, WARMup, FAIL, PASS, and COMPReq.

**Group** Compensation

**Syntax** COMPensate:STATus:MAInframe?

**Examples** COMPENSATE:STATUS:MAINFRAME? might return  
COMPENSATE:STATUS:MAINFRAME PASS, indicating that the current compensation data should allow the instrument to meet operating specifications.

## COMPensate:TEMPerature:CH<x>? (Query Only)

This query only command returns the difference (in °C) between the current temperature of the module channel and the associated temperature currently residing in the in-use run-time compensation memory.

**Group** Compensation

**Syntax** COMPensate:TEMPerature:CH<x>?

**Examples** COMPENSATE:TEMPERATURE:CH1? might return  
COMPENSATE:TEMPERATURE:CH1 1.5.

## COMPensate:TEMPerature:MAInframe? (Query Only)

This query only command returns the difference (in °C) between the current temperature of the mainframe and the associated temperature currently residing in the in-use run-time compensation memory.

**Group** Compensation

**Syntax** COMPensate:TEMPerature:MAInframe?

**Examples** COMPENSATE:TEMPERATURE:MAINFRAME? might return  
COMPENSATE:TEMPERATURE:MAINFRAME 2.7.

## CURSor?

This is a query only command that returns all of the current cursor settings.

**Group** Cursor

**Syntax** CURSor?

**Examples** CURSOR? might return CURSOR:CURSOR1:COLOR 255; SOURCE CH3,MAIN;  
CURSOR:CURSOR2:COLOR 255; SOURCE CH3,MAIN; CURSOR:DIV2  
0;FUNCTION WAVEFORMS; SELECT CURSOR1; HBAR:DELTA  
-0.401084010840; POSITION1 0.201897018970;POSITION2  
-0.199186991870; CURSOR:VBAR:DELTA 19.9398797595E-009;  
POSITION1 31.1302605210E-009; POSITION2 51.0701402806E-009;  
CURSOR:WAVEFORM:HDELTA 0.485961914062; VDELTA  
-19.9398797595E-009; HPOS1 0.240936279297; HPOS2  
-0.245025634766; POSITION1 31.1302605210E-009; POSITION2  
51.0701402806E-009 as the current cursor settings.

## CURSor:CURSor<x>:COLOR

This command sets or queries the color of the cursor specified by x, which can be 1 or 2. This command is the same as the DISplay:COLor:CURSOR<x> command. This is equivalent to setting Color in the Cursor 1 or Cursor 2 section of the Cursor Setup dialog box. The default cursor color is red.

The argument for the command is comprised of 3 fields (red, green, and blue). You specify the value for each of the three fields to specify the cursor color.

The following table lists values for common colors.

### Index number values for common cursor colors

Name	Index number
Black	0
Blue	16711680
Aqua	16776960
Lime	65280

**Index number values for common cursor colors (cont.)**

<b>Name</b>	<b>Index number</b>
Fuchsia	16711935
Red	255
Yellow	65535
White	16777215
Navy	8388608
Teal	8421376
Green	32768
Purple	8388736
Maroon	128
Olive	32896
Gray	8421504
Silver	12632193

**Group** Cursor

**Syntax** CURSOR:CURSOR<x>:COLOR <NR1>  
CURSOR:CURSOR<x>:COLOR?

**Related Commands** [DISplay:COLor:CURSor<x>](#)

**Arguments** NR1 is an integer in the range of 0 through 16777215 representing a color defined as follows: red + 256 × green + 65536 × blue, where red, green and blue are values between 0 and 255.

**Examples** CURSOR:CURSOR2:COLOR 255 sets the color of Cursor 2 to red.  
CURSOR:CURSOR1:COLOR? might return CURSOR:CURSOR1:COLOR 8421376, indicating that the color of Cursor 1 is Teal.

## **CURSOR:CURSOR<x>:SOURCE**

This command sets or queries which waveform is associated with the specified cursor. The cursor is specified by x, which can be 1 or 2. This command can optionally set the target timebase. This is equivalent to selecting a waveform (and timebase) in the Source control for the Cursor 1 or Cursor 2 section of the Cursor Setup dialog box. If you do not set the timebase, the default is Main for both cursors.



<b>Group</b>	Cursor
<b>Syntax</b>	<code>CURSOR:CURSOR&lt;x&gt;:SOURCE {CH&lt;x&gt;   MATH&lt;x&gt;   REF&lt;x&gt;}[, {MAIN   MAG1   MAG2}]</code> <code>CURSOR:CURSOR&lt;x&gt;:SOURCE?</code>
<b>Arguments</b>	<ul style="list-style-type: none"> <li>■ CH&lt;x&gt; selects a channel waveform as the source for the specified cursor. The range for x is 1 through 8.</li> <li>■ MATH&lt;x&gt; selects a math waveform as the source for the specified cursor. The range for x is 1 through 8.</li> <li>■ REF&lt;x&gt; selects a reference waveform as the source for the specified cursor. The range for x is 1 through 8.</li> <li>■ MAIN (optional) selects Main as the timebase for the specified cursor display.</li> <li>■ MAG1 (optional) selects MAG1 as the timebase for the specified cursor display.</li> <li>■ MAG2 (optional) selects MAG2 as the timebase for the specified cursor display.</li> </ul>
<b>Examples</b>	<p><code>CURSOR:CURSOR2:SOURCE CH8,MAG1</code> associates Cursor 2 with the channel 8 waveform and the MAG1 timebase.</p> <p><code>CURSOR:CURSOR1:SOURCE?</code> might return <code>CURSOR:CURSOR1:SOURCE REF2,MAIN</code>, indicating that Cursor 1 is associated with the REF2 waveform and the Main timebase.</p>

## CURSOR:FUNCTION

This command sets or queries cursor type. Cursors are attached to the selected waveform. Sending this command is equivalent to setting Function in the Cursor Setup dialog box.

<b>Group</b>	Cursor
<b>Syntax</b>	<code>CURSOR:FUNCTION { HBARS   OFF   VBARS   WAVEFORMS }</code> <code>CURSOR:FUNCTION?</code>

**Related Commands**    [SELECT:CONTROL](#)

- Arguments**
- **HBARs** enables horizontal bar cursors, which provide traditional vertical unit readouts for Cursor 1 (bar1), Cursor 2 (bar2), and the  $\Delta$  between them.
  - **OFF** removes the cursors from the display.
  - **VBARs** enables vertical bar cursors, which provide traditional horizontal unit readouts for Cursor 1 (bar1), Cursor 2 (bar2), the  $\Delta$  between them, and  $1/\Delta t$  (frequency when the horizontal unit is time),  $1/\Delta b$  (bits per second when the horizontal unit is bits) and  $\Delta d/2$  (computes the one way distance when the horizontal unit is distance)..
  - **WAVEforms** enables the waveform cursors, which provide both horizontal unit readouts and vertical unit readouts for Cursor 1 and Cursor 2, which are constrained to valid data points of the selected waveform.

**Examples**     `CURSOR:FUNCTION VBARS` enables the vertical bar type cursors.

`CURSOR:FUNCTION?` might return `CURSOR:FUNCTION HBARS`, indicating that the horizontal bar cursors are enabled.

## **CURSor:HBARs? (Query Only)**

This query only command returns the current settings for the horizontal bar cursors.

**Group**     Cursor

**Syntax**     `CURSor:HBARs?`

**Examples**     `CURSOR:HBARS?` might return `CURSOR:HBARS:DELTA`  
`-4.01084010840E-003; POSITION1 2.01897018970E-003;`  
`POSITION2 -1.99186991870E-003.`

## **CURSor:HBARs:DELTA? (Query Only)**

This query only command returns the difference between the two horizontal bar cursors.

**Group**     Cursor

**Syntax**     `CURSor:HBARs:DELTA?`

**Related Commands** [CURSor:FUNctIon](#)

**Returns** NR3

**Examples** CURSOR:HbARS:DELTA? might return CURSOR:HbARS:DELTA  
-3.00813008130E-003 for the difference between the two horizontal bar  
cursors.

## CURSor:HbARs:POSition<x>

This command sets or queries the position of a horizontal bar cursor, which is constrained to be visible in the selected timebase. (Use the CURSor:FUNctIon command to set the cursor function to horizontal bars.) The cursor is specified by x, which can be 1 or 2. This is equivalent to setting a value in the Position control in the Cursor 1 or Cursor 2 sections of the Cursor Setup dialog box (when Horizontal Bars is selected in the Function section).

**Group** Cursor

**Syntax** CURSor:HbARs:POSition<x> <NR3>  
CURSor:HbARs:POSition<x>?

**Related Commands** [CURSor:FUNctIon](#)

**Arguments** NR3 specifies the cursor position relative to zero for the source waveform.

**Examples** CURSOR:HbARS:POSITION1 3.0E-3 positions Cursor 1 at 3 mV units above the zero level of the source waveform.

CURSOR:HbARS:POSITION2? might return CURSOR:HbARS:POSITION2  
-1.96476964770E-003, indicating that Cursor 2 is approximately 1.965 mV below the zero level of the source waveform.

## CURSor:SElect

This command sets or queries which cursor is active for front-panel control. The active cursor is displayed as a solid line and can be moved using the front-panel general-purpose knob. The unselected cursor is displayed as a dashed line. This command is equivalent to clicking in the Position control box in the Cursor 1 or Cursor 2 section of the Cursor Setup dialog box.

<b>Group</b>	Cursor
<b>Syntax</b>	<code>CURSor:SElect { CURSOR1   CURSOR2   NONE }</code> <code>CURSor:SElect?</code>
<b>Arguments</b>	<ul style="list-style-type: none"> <li>■ CURSOR1 makes cursor 1 the active cursor.</li> <li>■ CURSOR2 makes cursor 2 the active cursor.</li> <li>■ NONE disables cursor 1 and cursor 2 so that they cannot be moved by the front-panel general-purpose knob.</li> </ul>
<b>Examples</b>	<p><code>CURSOR:SELECT CURSOR1</code> sets cursor 1 as the active cursor, which is controllable by the front-panel general-purpose knob.</p> <p><code>CURSOR:SELECT?</code> might return <code>CURSOR:SELECT CURSOR1</code>, indicating that cursor 1 is the active cursor and can be controlled by the front-panel general-purpose knob.</p>

## CURSor:VBArS? (Query Only)

This query only command returns the current settings of the vertical bar cursors.

<b>Group</b>	Cursor
<b>Syntax</b>	<code>CURSor:VBArS?</code>
<b>Examples</b>	<p><code>CURSOR:VBARS?</code> might return <code>CURSOR:VBARS:DELTA</code> <code>1.00601202405E-009; POSITION1 20.9968917836E-009;</code> <code>POSITION2 22.0029038076E-009.</code></p>

## CURSor:VBArS:DELTA? (Query Only)

This query only command returns the difference between the two vertical bar cursors. This corresponds to the  $\Delta t$  value in the Cursors readout.

<b>Group</b>	Cursor
<b>Syntax</b>	<code>CURSor:VBArS:DELTA?</code>

**Returns** NR3

**Examples** `CURSOR:VBARS:DELTA?` might return `CURSOR:VBARS:DELTA 1.65330661323E-009`, indicating the delta between the vertical bar cursors in absolute horizontal units.

## **CURSor:VBARs:POSition<x>**

This command sets or queries the position of the specified vertical bar cursor, which is constrained to be within the displayed range. The cursor is specified by *x*, which can be 1 or 2. (Use the CURSor FUNCtion command to set the cursor function to vertical bars.) This is equivalent to entering a value in the Cursor 1 or Cursor 2 Position control in the Cursor Setup dialog box.

**Group** Cursor

**Syntax** `CURSor:VBARs:POSition<x> <NR3>`  
`CURSor:VBARs:POSition<x>?`

**Related Commands** [HORizontal:MAIn:TOFPoint?](#), [HORizontal:MAGnify<x>:TOFPoint?](#), [HORizontal:MATH<x>:MAIn:TOFPoint?](#), [HORizontal:MATH<x>:MAGnify<x>:TOFPoint?](#), [HORizontal:REF<x>:MAIn:TOFPoint?](#), [HORizontal:REF<x>:MAGnify<x>:TOFPoint?](#)

**Arguments** NR3 specifies the cursor position measured from the trigger point of the source waveform.

**Examples** `CURSOR:VBARS:POSITION2 9.0E-6` positions the cursor 2 vertical bar cursor at 9  $\mu$ s from the trigger point of the source waveform.

`CURSOR:VBARS:POSITION1?` might return `CURSOR:VBARS:POSITION1 17.4929859719E-009`, indicating that the cursor 1 vertical bar is at 17.49 ns from the trigger point of the source waveform.

## **CURSor:WAVEform? (Query Only)**

This query only command returns all of the waveform cursor parameters.

**Group** Cursor

**Syntax** CURSor:WAVeform?

**Examples** CURSOR:WAVEFORM? might return CURSOR:WAVEFORM:HDELTA 0.487243652344;VDELTA -19.9398797595E-009; HPOS1 0.241729736328;HPOS2 -0.245513916016; POSITION1 31.1302605210E-009; POSITION2 51.0701402806E-009.

## CURSor:WAVeform:HDELTA? (Query Only)

This query-only command returns the horizontal difference between the waveform cursors. This is the absolute value of the horizontal position of the first cursor minus the horizontal position of the second cursor. This is equivalent to the waveform  $\Delta t$  readout value.

**Group** Cursor

**Syntax** CURSor:WAVeform:HDELTA?

**Examples** CURSOR:WAVEFORM:HDELTA? might return CURSOR:WAVEFORM:HDELTA 1.03E-9, indicating that the time between the waveform cursors is 1.03 ns.

## CURSor:WAVeform:HPOS<x>? (Query Only)

This query only command returns the position of the specified waveform cursor. The cursor is specified by x, which can be 1 or 2. This corresponds to the v1 or v2 (cursor 1 or cursor 2) cursor readout.

**Group** Cursor

**Syntax** CURSor:WAVeform:HPOS<x>?

**Examples** CURSOR:WAVEFORM:HPOS1? might return CURSOR:WAVEFORM:HPOS2 0.247344970703, indicating that cursor 2 is at 247.3 mV with respect to ground on the source waveform.

## CURSor:WAVeform:POStion<x>

This command sets or queries the position of a waveform cursor, which is constrained to be visible in the selected timebase. (Use the CURSor FUNCtion

command to set the cursor function to Waveform.) The cursor is specified by x, which can be 1 or 2. This is equivalent to setting a value in the Position control in the Cursor 1 or Cursor 2 sections of the Cursor Setup dialog box (when Waveform is selected in the Function section).

<b>Group</b>	Cursor
<b>Syntax</b>	CURSOR:WAVEform:POSition<x> <NR3> CURSOR:WAVEform:POSition<x>?
<b>Related Commands</b>	<a href="#">CURSOR?</a>
<b>Arguments</b>	NR3 specifies the cursor position measured relative to the time of the trigger point of the source waveform.
<b>Examples</b>	CURSOR:WAVEFORM:POSITION2 9.00E-6 positions the cursor 2 waveform cursor at 9 $\mu$ s.  CURSOR:WAVEFORM:POSITION2? might return CURSOR:WAVEFORM:POSITION2 51.0701402806E-009, indicating waveform cursor 1 is at 51.1 ns.

## CURSOR:WAVEform:VDELTA? (Query Only)

This query only command returns the vertical difference between the waveform cursors. This is the absolute value of the vertical position of the first cursor minus the vertical position of the second cursor. This is equivalent to the waveform  $\Delta v$  readout value.

<b>Group</b>	Cursor
<b>Syntax</b>	CURSOR:WAVEform:VDELTA?
<b>Examples</b>	CURSOR:WAVEFORM:VDELTA? might return CURSOR:WAVEFORM:VDELTA 1.06E-3, indicating that the voltage between the waveform cursors is 1.06 mV.

## CURVe

The query form of this command transfers waveform data from the instrument in binary or ASCII format. Use the DATA:SOURCE and DATA:TIMEbase commands

to specify the location (source) of the waveform data that is transferred from the instrument. Each waveform that is transferred has an associated waveform preamble that contains information such as data format and scale. Refer to the [WFMOupre?](#) query for information about the waveform preamble after the curve query command is completed. The data format is specified by the `DATA:ENCdg` command.

The first and last data points that are transferred are specified by the `DATA:START` and `DATA:STOP` commands.

The `CURVe` command transfers waveform data to the instrument. The data is stored in the reference memory location specified by `DATA:DESTination`, starting with the data point specified by the `DATA:START` command. Only one waveform can be transferred at a time. If the destination reference is not empty, the reference slot will be cleared, and the instrument will attempt to create a waveform using the provided data points. The waveform will only be displayed if the reference is displayed.

**Group** Waveform Transfer

**Syntax** `CURVe {<block> | <ascii>}`  
`CURVe?`

**Related Commands** [DATA:SOURce](#), [DATA:TIMEbase](#), [WFMOupre?](#), [DATA:ENCdg](#), [DATA:START](#), [DATA:STOP](#), [DATA:DESTination](#)

- Arguments**
- `block` is the waveform data in binary format. The waveform is formatted as: `#<x><yyy><data><newline>` where `<x>` is the number of `y` bytes. For example, if `<yyy> = 500`, then `<x> = 3`. `<yyy>` is the number of bytes to transfer including checksum.
  - `ascii` is the waveform data in ASCII format. The format for ASCII data is `<NR1>[,<NR1>...]` where each `<NR1>` represents a data point.

**Examples** `CURVe?` might return (partial)

```
CURVE -509476864,-512163840, -511115264, -510918656,
-516161536,-484179968, -117112832,207093760,
322437120,351600640,409206784,468451328,
482934784,490668032,495517696,432668672,
482148352,483655680,493617152,495321088,
499843072,493420544,488964096,497221632,
503250944,501088256,501743616,503447552,
498139136,502857728,507248640,507117568,
503054336,506200064,505413632,502988800,
506658816,506920960,498008064,501415936,
```



504299520, 498991104, 496304128, 499056640,  
 501612544, 505806848, 498794496, 494206976, 414449664,  
 -11141120, -274071552, -394330112, -432865280, -450756608,  
 -488439808, -500826112. ...

## DATA

This command sets or queries the format and location of the waveform data that is transferred with the CURVe command.

**Group** Waveform Transfer

**Syntax** DATA { INIT | SNAP }  
 DATA?

**Related Commands** [CURVe](#), [WAVFrm?](#), [DATA:START](#), [DATA:STOP](#)

- Arguments**
- INIT initializes the waveform data parameters to their factory defaults.
  - SNAP sets DATA:START and DATA:STOP to match the current vertical bar cursor positions.

**Examples** DATA SNAP assigns the current position of the vertical bar cursors to DATA:START and DATA:STOP.

DATA? might return DATA:DESTINATION REF1;SOURCE CH3;ENCDG ASCII;START 1;STOP 500.

## DATA:DESTINATION

This command sets or queries the destination reference memory location for storing waveform data that is transferred into the instrument by the CURVe command.

**Group** Waveform Transfer

**Syntax** DATA:DESTINATION {REF<x>}  
 DATA:DESTINATION?

**Related Commands** [CURVe](#), [WAVFrm?](#), [DATA:START](#), [DATA:STOP](#)

**Arguments** REF<X> REF1 through REF8 are the reference memory locations where the waveform will be stored. You cannot have a channel as a destination.

**Examples** DATA:DESTINATION REF3 stores the incoming waveform data in Reference 3.  
 DATA:DESTINATION? might return DATA:DESTINATION REF8 as the reference memory location that is currently selected.

## DATA:ENCdg

This command sets or queries the format of the waveform data. This command is equivalent to setting WFMOutpre:ENCdg, WFMOutpre:BN\_Fmt, and WFMOutpre:BYT\_Or, as shown in the table below. Only signed, 4-byte integers and floating point values are supported. Changing the DATA:ENCdg parameter will coerce WFMOutpre formatter commands accordingly. If ASCII is selected, the output will be formatted to floating point or signed integer, based on how the data is stored in the instrument.

**Group** Waveform Transfer

**Syntax** DATA:ENCdg { ASCII | RIBinary | FPBinary | SRIBinary | SFPBinary }  
 DATA:ENCdg?

**Related Commands** [WFMOutpre:ENCdg](#), [WFMOutpre:BN\\_Fmt](#), [WFMOutpre:BYT\\_Or](#)

- Arguments**
- ASCII specifies the ASCII representation of signed integer (RIBinary) data if the data source is a live channel; otherwise, if the data source is equal to the math then the ASCII curve data is in floating point format. If this is the value at power on, the WFMOutpre values for BN\_Fmt, BYT\_Or, and ENCdg are set as RI, MSB, and ASCII respectively.
  - RIBinary specifies signed integer data-point representation with the most-significant byte transferred first.
  - FPBinary specifies floating-point data-point representation with the most-significant byte transferred first.
  - SRIBinary is the same as RIBinary except that the byte order is reversed; the least-significant byte is transferred first. This format is useful when transferring data to IBM-compatible PCs.
  - SFPBinary is the same as FPBinary except that the byte order is reversed; the least-significant byte is transferred first. This format is useful when transferring data to IBM-compatible PCs.

- Examples** `DATA:ENCDG RIBINARY` sets the data-encoding format to signed integer, with the most-significant byte transferred first.
- `DATA:ENCDG?` might return `DATA:ENCDG SFPBINARY`, indicating that the format of the data is floating point, with the least-significant bit transferred first.

## DATA:SOURce

This command sets or queries the location of the waveform data that is transferred from the instrument by the `CURVe?` query. Only one source can be set at a time. Changing the source waveform does not cause the `WFMOutpre` interpretation commands to update. To update these commands, you must send the `CURVe?` command. You may also set or query the timebase associated with the waveform data transferred from the instrument using the `DATA:TIMEbase` command.

**Group** Waveform Transfer

**Syntax** `DATA:SOURce { CH<x> | MATH<x> | REF<x> }`  
`DATA:SOURce?`

**Related Commands** [DATA:TIMEbase](#)

- Arguments**
- `CH<x>` selects the specified channel (1 through 8) acquisition waveform as the waveform source.
  - `MATH<x>` selects the specified Math (1 through 8) waveform as the waveform source.
  - `REF<x>` selects the specified Reference (1 through 8) waveform as the waveform source.

- Examples** `DATA:SOURce REF2` specifies that the Reference 2 waveform will be transferred in the next `CURVe?` query.
- `DATA:SOURce?` might return `DATA:SOURce REF3`, indicating that Reference 3 waveform is the source for the waveform data that is transferred using a `CURVe?` query.

## DATA:STARt

Sets or queries the starting data point for waveform transfer. This command allows for the transfer of partial waveforms to and from the instrument.

<b>Group</b>	Waveform Transfer
<b>Syntax</b>	DATA:START <NR1> DATA:START?
<b>Related Commands</b>	<a href="#">CURVe</a> , <a href="#">DATA:STOP</a>
<b>Arguments</b>	NR1  This value ranges from 1 to the record length and is the first data point that will be transferred. Data will be transferred from this point to DATA:STOP or the record length, whichever is less. If this value is greater than the record length, then no data will be transferred.  When DATA:STOP is less than DATA:START, the stop value will equal DATA:START + (DATA:START - DATA:STOP). For example, if DATA:START = 30 and DATA:STOP = 20, then the range of data points for the waveform transfer will equal 30 through 40.
<b>Examples</b>	DATA:START 10 specifies that the waveform transfer will begin with data point 10.  DATA:START? might return DATA:START 214, indicating that 214 is the first waveform data point that will be transferred.

## DATA:STOP

Sets or queries the last data point that will be transferred when using the CURVe? query. This allows the transfer of partial waveforms to the controller.

When using the CURVe command, the instrument will stop reading data when there is no more data to read or when the specified record length is reached. When the specified record length is reached, this command will be ignored.

<b>Group</b>	Waveform Transfer
<b>Syntax</b>	DATA:STOP <NR1> DATA:STOP?
<b>Related Commands</b>	<a href="#">CURVe</a> , <a href="#">DATA:START</a>

- Arguments** NR1 ranges from 1 to the record length and is the last data point that will be transferred. If this value is greater than the record length, then data will be transferred up to the record length. If both DATA:START and DATA:STOP are greater than the record length, an execution error will be generated.
- If you always want to transfer complete waveforms, just set DATA:START to 1 and DATA:STOP to the maximum record length.
- When DATA:STOP is less than DATA:START, the stop value will equal DATA:START + (DATA:START - DATA:STOP). For example, if DATA:START = 30 and DATA:STOP = 20, then the range of data points for the waveform transfer will equal 30 through 40.
- Examples** DATA:STOP 4000 specifies that the waveform transfer will stop at data point 4000.
- DATA:STOP? might return DATA:STOP 500, indicating that 500 is the last data point that will be transferred.

## DATA:TIMEbase

Sets or queries the DATA:SOURce associated timebase used to specify the location of the waveform data to be transferred from the instrument when using the CURVe? command. The specified timebase must be active and displayed in order to receive data.

**Group** Waveform Transfer

**Syntax** DATA:TIMEbase { MAIN | MAG1 | MAG2 }  
DATA:TIMEbase?

**Related Commands** [CURVe](#), [DATA:SOURce](#)

- Arguments**
- MAIN selects MAIN as the timebase for the waveform data to be transferred out by the next CURVe? command.
  - MAG1 selects MAG1 as the timebase for the waveform data to be transferred out by the next CURVe? Command.
  - MAG2 selects MAG2 as the timebase for the waveform data to be transferred out by the next CURVe? command.

**Examples** DATA:TIMEBASE MAG1 sets the timebase for the waveform data to be transferred out to MAG1.

DATA:TIMEBASE? might return DATA:TIMEBASE MAIN, indicating that the timebase for the source waveform data is set to main.

## DATE

This command sets or queries the date that the instrument can display.

**Group** Miscellaneous

**Syntax** DATE <QString>  
DATE?

- Arguments**
- <QString> is a date in the form "yyyy-mm-dd".
  - yyyy is a four-digit year number.
  - mm is a two-digit month number from 01 to 12.
  - dd is a two-digit day number in the month.

---

**NOTE.** *There must a dash (-) after the yyyy and after the mm.*

---

**Examples** DATE "2006-01-24" sets the date to January 24, 2006.  
DATE? might return DATE 2006-06-01, indicating that the date is June 1, 2006.

## DELEte:WAVEform (No Query Form)

This command (no query form) deletes one or all of the stored reference waveforms from memory. The memory allocated for the reference location is then available for reallocation.

---

**NOTE.** *A reference waveform cannot be deleted if it is a source client for some other operation, such as a histogram source, math source, measurement, or waveform database.*

---

**Group** Save and Recall

**Syntax** DELEte:WAVEform { REF<x> | ALL }

**Related Commands** [RECAI:WAVEform](#), [SAVe:WAVEform](#)

- Arguments**
- REF<x> specifies one of the reference memory locations (REF1 through REF8).
  - ALL specifies all the stored reference waveforms.

**Examples** DELETE:WAVEFORM ALL removes all the waveforms stored in reference memory.  
DELETE:WAVEFORM REF2 removes the waveform stored at REF2.

## DESE

This command sets or queries the bits in the Device Event Status Enable Register (DESER). The DESER is the mask that determines whether events are reported to the Standard Event Status Register (SESR) and entered into the Event Queue. For a more detailed discussion of the use of these registers, see Registers for more information.

**Group** Status and Error

**Syntax** DESE <NR1>  
DESE?

**Related Commands** [\\*CLS](#), [\\*ESE](#), [\\*ESR?](#), [EVENT?](#), [EVMsg?](#), [\\*SRE](#), [\\*STB?](#)

**Arguments** <NR1>

This is a value in the range from 0 through 255. The binary bits of the DESER are set according to this value. For example, DESE 209 sets the DESER to the binary value 11010001 (that is, the most significant bit in the register is set to 1, the next most significant bit to 1, the next bit to 0, and so on).

The power-on default for DESER is all bits set if \*PSC is 1. If \*PSC is 0, the DESER maintains its value through a power cycle.

---

**NOTE.** *Setting the DESER and the ESER to the same value allows only those codes to be entered into the Event Queue and summarized on the ESB bit (bit 5) of the Status Byte Register. Use the \*ESE command to set the ESER.*

---

**Examples** DESE 209 sets the DESER to binary 11010001, which enables the PON, URQ, EXE, and OPC bits.

DESE? might return DESE 186, indicating that the DESER contains the binary value 10111010.

## DISplay? (Query Only)

This is a query only command that returns the current display settings.

**Group** Display Control

**Syntax** DISplay?

**Examples** DISPLAY? might return the following:

```
DISPLAY:PERSISTENCE 0.200000000000; STYLE NORMAL; INTERPOLAT
SINX;SHOWVECTOR 1;GRATICULE:STYLE FULL; HDIVS 1.000000000000;
VDIVS 1.000000000000; :DISPLAY:COLOR:BACKGROUND 0; FOREGROUND
12632193;CURSOR1 255; CURSOR2 255; HISTOGRAM 16776960; MASK
16711935; :DISPLAY:MEASREADOUT 0; CURSREADOUT 0; HISTREADOUT
0;WFMREADOUT 1; MASKREADOUT 0; MEASBAR 1.
```

## DISplay:COLor? (Query Only)

This query only command returns the colors used in the display. These include colors assigned to the following:

- Graticule background
- Graticule foreground
- Histogram
- Mask
- Cursor 1
- Cursor 2

The following table lists values for common colors.

### Index number values for common cursor colors

Name	Index number
Black	0
Blue	16711680
Aqua	16776960
Lime	65280



**Index number values for common cursor colors (cont.)**

<b>Name</b>	<b>Index number</b>
Fuchsia	16711935
Red	255
Yellow	65535
White	16777215
Navy	8388608
Teal	8421376
Green	32768
Purple	8388736
Maroon	128
Olive	32896
Gray	8421504
Silver	12632193

**Group** Display Control

**Syntax** DISPLAY:COLOR?

**Examples** DISPLAY:COLOR? might return DISPLAY:COLOR:BACKGROUND 0; FOREGROUND 12632193; CURSOR1 255; CURSOR2 255; HISTOGRAM 16776960; MASK 16711935.

## DISplay:COLor:BACKground

This command sets or queries the color of the graticule background. This is equivalent to setting the Background color in the Graticule section of the Display Setup dialog box. The default background color is black.

The argument for the command is an integer equal to the sum of three color values (red, green, and blue). You specify the value for each of the three color values and sum them to specify the background color.

The following table lists values for common colors.

**Index number values for common colors**

<b>Name</b>	<b>Index number</b>
Black	0
Blue	16711680
Aqua	16776960

**Index number values for common colors (cont.)**

<b>Name</b>	<b>Index number</b>
Lime	65280
Fuchsia	16711935
Red	255
Yellow	65535
White	16777215
Navy	8388608
Teal	8421376
Green	32768
Purple	8388736
Maroon	128
Olive	32896
Gray	8421504
Silver	12632193

**Group** Display Control

**Syntax** DISplay:COLor:BACKground <NR1>  
DISplay:COLor:BACKground?

**Arguments** NR1 is an integer in the range of 0 through 16777215 representing a color defined as follows:  $\text{red} + 256 \times \text{green} + 65536 \times \text{blue}$  where red, green and blue are values between 0 and 255.

**Examples** DISPLAY:COLOR:BACKGROUND 16776960 sets the color of the graticule background to aqua.  
  
DISPLAY:COLOR:BACKGROUND? might return DISPLAY:COLOR:BACKGROUND 016777215, indicating that the color of the graticule background is white.

## DISplay:COLor:CURSor<x>

This command sets or queries the color of the specified cursor (x), which may be 1 or 2. This is equivalent to setting Color in the Cursor 1 or Cursor 2 section of the Cursor Setup dialog box. The default cursor color is red.

The following table lists values for common colors.

**Index number values for common cursor colors**

<b>Name</b>	<b>Index number</b>
Black	0
Blue	16711680
Aqua	16776960
Lime	65280
Fuchsia	16711935
Red	255
Yellow	65535
White	16777215
Navy	8388608
Teal	8421376
Green	32768
Purple	8388736
Maroon	128
Olive	32896
Gray	8421504
Silver	12632193

**Group** Display Control

**Syntax** `DISPlay:COLor:CURSor<x> <NR1>`  
`DISPlay:COLor:CURSor<x>?`

**Related Commands** [CURSor:CURSor<x>:COLOR](#)

**Arguments** NR1 is an integer in the range of 0 through 16777215 representing a color defined as follows:  $\text{red} + 256 \times \text{green} + 65536 \times \text{blue}$  where red, green and blue are values between 0 and 255.

**Examples** `DISPLAY:COLOR:CURSOR1 128` sets the color of cursor 1 to maroon.  
`DISPLAY:COLOR:CURSOR1?` might return `DISPLAY:COLOR:CURSOR1 8421504`, indicating that the color of cursor 1 is gray.

## DISPlay:COLor:FOREground

This command sets or queries the color of the graticule foreground, which is the color of the graticule itself. This is equivalent to setting the Foreground color in

the Graticule section of the Display Setup dialog box. The default foreground color is silver.

The argument for the command is an integer equal to the sum of three color values (red, green, and blue). You specify the value for each of the three color values and sum them to specify the background color.

The following table lists values for common colors.

**Index number values for common colors**

<b>Name</b>	<b>Index number</b>
Black	0
Blue	16711680
Aqua	16776960
Lime	65280
Fuchsia	16711935
Red	255
Yellow	65535
White	16777215
Navy	8388608
Teal	8421376
Green	32768
Purple	8388736
Maroon	128
Olive	32896
Gray	8421504
Silver	12632193

**Group** Display Control

**Syntax** DISPLAY:COLOR:FOREGROUND <NR1>  
 DISPLAY:COLOR:FOREGROUND?

**Arguments** NR1 is an integer in the range of 0 through 16777215 representing a color defined as follows:  $\text{red} + 256 \times \text{green} + 65536 \times \text{blue}$  where red, green and blue are values between 0 and 255.

**Examples** DISPLAY:COLOR:FOREGROUND 255 sets the color of the foreground (graticule) to red.

DISPLAY:COLOR:FOREGROUND? might return DISPLAY:COLOR:FOREGROUND 16777215, indicating that the color of the foreground (graticule) is white.

## DISplay:COLor:HIStogram

This command sets or queries the color of the histogram data and its boundary area. (This command is the same as the HISTogram:COLOR command.) This is equivalent to setting Color in the Display Options section of the Histogram Setup dialog box. The default histogram color is aqua.

The argument for the command is an integer equal to the sum of three color values (red, green, and blue). You specify the value for each of the three color values and sum them to specify the background color.

The following table lists values for common colors.

### Index number values for common colors

Name	Index number
Black	0
Blue	16711680
Aqua	16776960
Lime	65280
Fuchsia	16711935
Red	255
Yellow	65535
White	16777215
Navy	8388608
Teal	8421376
Green	32768
Purple	8388736
Maroon	128
Olive	32896
Gray	8421504
Silver	12632193

**Group** Display Control

**Syntax** DISplay:COLor:HIStogram <NR1>  
DISplay:COLor:HIStogram?

**Related Commands** [HIStogram:COLOr](#)

**Arguments** NR1 is an integer in the range of 0 through 16777215 representing a color defined as follows:  $\text{red} + 256 \times \text{green} + 65536 \times \text{blue}$  where red, green and blue are values between 0 and 255.

**Examples**    `DISPLAY:COLOR:HISTOGRAM 128` sets the color of the histogram data and its boundary area to maroon.

`DISPLAY:COLOR:HISTOGRAM?` might return `DISPLAY:COLOR:HISTOGRAM 8421504`, indicating that the color of the histogram data and its boundary is gray.

## DISplay:COLor:MASK

This command sets or queries the color of the mask. (The command is the same as the `MASK:COLOr` command. This is equivalent to setting Color in the Options section of the Mask Setup dialog box. The default mask polygon color is fuchsia.

The argument for the command is an integer equal to the sum of three color values (red, green, and blue). You specify the value for each of the three color values and sum them to specify the background color.

The following table lists values for common colors.

### Index number values for common colors

Name	Index number
Black	0
Blue	16711680
Aqua	16776960
Lime	65280
Fuchsia	16711935
Red	255
Yellow	65535
White	16777215
Navy	8388608
Teal	8421376
Green	32768
Purple	8388736
Maroon	128
Olive	32896
Gray	8421504
Silver	12632193

**Group**    Display Control

**Syntax**    `DISplay:COLor:MASK <NR1>`  
`DISplay:COLor:MASK?`

**Related Commands**    [MASK:COLOr](#)

**Arguments**    NR1 is an integer in the range of 0 through 16777215 representing a color defined as follows:  $\text{red} + 256 \times \text{green} + 65536 \times \text{blue}$  where red, green and blue are values between 0 and 255.

**Examples**    `DISPLAY:COLOR:MASK 16711935` sets the color of the mask to fuchsia.  
`DISPLAY:COLOR:MASK?` might return `DISPLAY:COLOR:MASK 32768`, indicating that the color of the mask is green.

## DISplay:CURSReadout

This command sets or queries state of the Cursors readout. This is equivalent to checking or clearing the Cursor readout item in the readout selection menu.

**Group**    Display Control

**Syntax**    `DISplay:CURSreadout { ON | OFF | NR1 }`  
`DISplay:CURSreadout?`

**Arguments**

- ON enables the readout display for the Cursor function.
- OFF disables the readout display for the Cursor function.
- NR1 set to 0 disables the Cursor readout display; any other integer enables the Cursor readout display.

**Examples**    `DISPLAY:CURSREADOUT 1` enables the Cursor readout display.  
`DISPLAY:CURSREADOUT?` might return `DISPLAY:CURSREADOUT 0`, indicating that the Cursor readout display is off.

---

**NOTE.** *A query always returns a 0 or 1.*

---

## DISplay:DATE

This command sets or queries state of the Date/Time display. This is equivalent to toggling the Date/Time display on or off from the View menu.

**Group**    Display Control

**Syntax**    `DISplay:DATE { ON | OFF | NR1 }`  
`DISplay:DATE?`

- Arguments**
- ON turns on the Date/Time display, which appears in the lower-right corner of the instrument screen.
  - OFF turns off the Date/Time display.
  - NR1 set to 0 turns off the Date/Time display; any other integer turns it on.

**Examples**    `DISPLAY:DATE ON` enables the Date/Time display.  
`DISPLAY:DATE?` might return `DISPLAY:DATE 0`, indicating that the Date/Time display is off.

---

**NOTE.** *A query always returns a 0 or 1.*

---

## DISplay:GRAticule? (Query Only)

This query only command returns all the graticule parameters.

**Group**    Display Control

**Syntax**    `DISplay:GRAticule?`

**Examples**    `DISPLAY:GRATICULE?` might return `DISPLAY:GRATICULE:STYLE FULL;`  
`HDIVS 10;VDIVS 10.`

## DISplay:GRAticule:HDIVS? (Query Only)

This query only command returns the number of horizontal divisions in the graticule.

**Group**    Display Control

**Syntax**    `DISplay:GRAticule:HDIVS?`

**Examples**    `DISPLAY:GRATICULE:HDIVS?` might return `DISPLAY:GRATICULE:HDIVS 10`, indicating that the graticule display is 10 horizontal divisions wide.



## DISplay:GRAticule:STYLE

This command selects the style of graticule that is displayed. Sending the command is equivalent to setting Style in the Graticule section of the Display Setup dialog box.

**Group** Display Control

**Syntax** DISplay:GRAticule:STYLE { FULL | CROSSHair | GRID | FRAME }  
DISplay:GRAticule:STYLE?

- Arguments**
- FULL specifies a frame, a grid, and cross hairs.
  - CROSSHair specifies a frame and cross hairs.
  - GRID specifies a frame and a grid.
  - FRAME specifies just a frame.

**Examples** DISPLAY:GRATICULE:STYLE GRID sets the graticule style to display a frame and a grid.

DISPLAY:GRATICULE:STYLE? might return DISPLAY:GRATICULE:STYLE FULL when all graticule elements (grid, frame, and cross hairs) are displayed.

## DISplay:GRAticule:VDIVS? (Query Only)

This query only command returns the number of vertical divisions in the graticule.

**Group** Display Control

**Syntax** DISplay:GRAticule:VDIVS?

**Examples** DISPLAY:GRATICULE:VDIVS? might return DISPLAY:GRATICULE:VDIVS 10, indicating that the graticule display is 10 vertical divisions high.

## DISplay:HISTReadout

This command sets or queries state of the Histogram readout. This is equivalent to checking or clearing the Histogram readout item in the readout selection menu.

<b>Group</b>	Display Control
<b>Syntax</b>	DISplay:HISTReadout { ON   OFF   NR1 } DISplay:HISTReadout?
<b>Arguments</b>	<ul style="list-style-type: none"> <li>■ ON enables the readout display for the Histogram function.</li> <li>■ OFF disables the readout display for the Histogram function.</li> <li>■ NR1 set to 0 disables the Histogram readout display; any other integer enables the Histogram readout display.</li> </ul>

**Examples**    DISPLAY:HISTREADOUT 1 enables the Histogram readout display.  
 DISPLAY:HISTREADOUT? might return DISPLAY:HISTREADOUT 0, indicating that the Histogram readout display is off.

---

**NOTE.** *A query always returns a 0 or 1.*

---

## DISplay:INTERPolat

This command sets or queries the interpolation algorithm used to display any waveforms in the Main or Mag views when the record length is less than 1000 points. This is equivalent to setting Interpolation in the Style section of the Display Setup dialog box.

---

**NOTE.** *Interpolation is a display function only; the instrument does not do acquisition interpolation.*

---

With interpolation, the instrument increases the sample density on the waveforms it displays by calculating intermediate points for those sampled. You can set interpolation display algorithms to None, Linear, or Sin (x)/x.

<b>Group</b>	Display Control
<b>Syntax</b>	DISplay:INTERPolat { LINEar   NONE   SINX } DISplay:INTERPolat?
<b>Related Commands</b>	<a href="#">DISplay:STYle</a>

- Arguments**
- **LINEAR** specifies linear interpolation. This algorithm computes points between actual acquired samples by using a straight line fit. The algorithm assumes all interpolated points fall along the straight line. Linear interpolation is useful for many waveforms such as pulse trains.
  - **NONE** turns off the interpolation function.
  - **SINX** specifies Sin (x)/x interpolation. This algorithm computes points using a curve fit between the actual values acquired. It assumes all interpolated points fall along the curve. This is useful when displaying more rounded waveforms such as sine waves. This algorithm can be used for general use, but it may introduce some overshoot or undershoot in signals with fast rise times.

**Examples** `DISPLAY:INTERPOLAT LINEAR` selects the linear interpolation algorithm.

`DISPLAY:INTERPOLAT?` might return `DISPLAY:INTERPOLAT NONE`, indicating that no interpolation algorithm is selected.

## DISplay:MASKReadout

This command sets or queries state of the Mask readout. This is equivalent to checking or clearing the Mask readout item in the readout selection menu.

**Group** Display Control

**Syntax** `DISplay:MASKReadout { ON | OFF | NR1 }`  
`DISplay:MASKReadout?`

- Arguments**
- **ON** enables the readout display for the Mask function.
  - **OFF** disables the readout display for the Mask function.
  - **NR1** set to 0 disables the Mask readout display; any other integer enables the Mask readout display.

**Examples** `DISPLAY:MASKREADOUT 1` enables the Mask readout display.

`DISPLAY:MASKREADOUT?` might return `DISPLAY:MASKREADOUT 0`, indicating that the Mask readout display is off.

---

**NOTE.** *A query always returns a 0 or 1.*

---

## DISplay:MEASBar

This command sets or queries whether or not the Measurement Bar is displayed. This is equivalent to checking or clearing the Measurement Bar menu item in the View menu.

**Group** Display Control

**Syntax** DISplay:MEASBar { ON | OFF | NR1 }  
DISplay:MEASBar?

- Arguments**
- ON enables the Measurement Bar display.
  - OFF disables the Measurement Bar display.
  - NR1 set to 0 disables the Measurement Bar display; any other integer enables the Measurement Bar display.

**Examples** DISPLAY:MEASBAR 1 enables the Measurement Bar display.  
DISPLAY:MEASBAR? might return DISPLAY:MEASBAR 0, indicating that the Measurement Bar display is off.

---

**NOTE.** A query always returns a 0 or 1.

---

## DISplay:MEASReadout

This command sets or queries state of the measurement readout. This is equivalent to checking or clearing the Measurement readout item in the readout selection menu.

**Group** Display Control

**Syntax** DISplay:MEASReadout { ON | OFF | NR1 }  
DISplay:MEASReadout?

- Arguments**
- ON enables the readout display for the Measurement function.
  - OFF disables the readout display for the Measurement function.
  - NR1 set to 0 disables the Measurement readout display; any other integer enables the Measurement readout display.

**Examples** `DISPLAY:MEASREADOUT 1` enables the Measurement readout display.

`DISPLAY:MEASREADOUT?` might return `DISPLAY:MEASREADOUT 0`, indicating that the Measurement readout display is off.

---

**NOTE.** *A query always returns a 0 or 1.*

---

## DISplay:PERSiStence

This command sets or queries the length of time that data points are displayed when variable persistence is active. This affects the display only and is equivalent to setting Time in the Style section of the Display Setup dialog box.

**Group** Display Control

**Syntax** `DISplay:PERSiStence <NR3>`  
`DISplay:PERSiStence?`

**Related Commands** [DISplay:STyLe](#)

**Arguments** NR3 specifies the length, in seconds, that the waveform points are displayed on the screen. The range is 200 ms to 10 s (in increments of 100 ms from 200 ms to 1s, and in 0.5 s increments from 1 s to 10 s).

**Examples** `DISPLAY:PERSISTENCE 3` specifies that the waveform points are displayed on the screen for 3 seconds before they fade.

`DISPLAY:PERSISTENCE?` might return `DISPLAY:PERSISTENCE 0.200000000000`, indicating that the persistence is set to 200 ms.

## DISplay:SHOVVector

This command sets or queries the Show Vectors mode. For the Normal display style only, you can set the instrument to draw vectors between displayed waveform points. This is equivalent to selecting or clearing the Show Vectors check box in the Style section of the Display Setup dialog box.

**Group** Display Control

**Syntax** `DISplay:SHOVVector { ON | OFF | NR1 }`  
`DISplay:SHOVVector?`

**Related Commands**    [DISplay:STyLe](#)

- Arguments**
- ON enables the vector display.
  - OFF disables the vector display.
  - NR1 set to 0 disables the vector display; any other integer enables the vector display.

**Examples**    `DISPLAY:SHOWVECTOR ON` enables the Show Vectors mode, which displays lines between the sampled (or interpolated) points.

`DISPLAY:SHOWVECTOR?` might return `DISPLAY:SHOWVECTOR 0`, indicating that the vector display is disabled, and the waveform is displayed as dots.

---

**NOTE.** *A query always returns a 0 or 1.*

---

## DISplay:STyLe

This command sets or queries whether or not the data is displayed using persistence. Sending this command is equivalent to selecting Variable Persistence or Infinite Persistence (or neither) in the Style section of the Display Setup dialog box.

**Group**    Display Control

**Syntax**    `DISplay:STyLe { NORMal | INFPersist | VARPersist }`  
`DISplay:STyLe?`

**Related Commands**    [DISplay:PERsistence](#)

- Arguments**
- `NORMal` disables the persistence mode.
  - `INFPersist` accumulates data points on the display indefinitely. The display is reset when the style or acquisition is reset.
  - `VARPersist` leaves acquired data points on the display for a period of time specified by the `DISplay:PERsistence` command.

**Examples**    `DISPLAY:STYLE INFPERSIST` sets the display to indefinitely accumulate data points on the screen.

DISPLAY:STYLE? might return DISPLAY:STYLE VARPERSIST, indicating that the display style is set to variable persistence.

## DISplay:WFMReadout

This command sets or queries state of the Waveform readout. This is equivalent to checking or clearing the Waveform readout item in the readout selection menu.

**Group** Display Control

**Syntax** DISplay:WFMReadout { ON | OFF | NR1 }  
DISplay:WFMReadout?

- Arguments**
- ON enables the readout display for the Waveform function.
  - OFF disables the readout display for the Waveform function.
  - NR1 set to 0 disables the Waveform readout display; any other integer enables the Waveform readout display.

**Examples** DISPLAY:WFMREADOUT 1 enables the Waveform readout display.

DISPLAY:WFMREADOUT? might return DISPLAY:WFMREADOUT 0, indicating that the Waveform readout display is off.

---

**NOTE.** *A query always returns a 0 or 1.*

---

## \*ESE

This command sets or queries the bits in the Event Status Enable Register (ESER). The ESER prevents events from being reported to the Status Byte Register (SBR). For a more detailed discussion of the use of these registers, see Registers.

**Group** Status and Error

**Syntax** \*ESE <NR1>  
\*ESE?

**Related Commands** \*CLS, DESE, \*ESR?, EVENT?, EVMsg?, \*SRE, \*STB?

**Arguments** NR1 specifies the binary bits of the ESER according to this value, which ranges from 0 through 255.

The power-on default for ESER is 0 if \*PSC is 1. If \*PSC is 0, the ESER maintains its value through a power cycle.

---

**NOTE.** *Setting the DESER and the ESER to the same value allows only those codes to be entered into the Event Queue and summarized on the ESB bit (bit 5) of the Status Byte Register. Use the DESE command to set the DESER. See Event Handling Sequence for a discussion of event handling.*

---

**Examples** \*ESE 209 sets the ESER to binary 11010001, which enables the PON, URQ, EXE, and OPC bits.

\*ESE? might return the string \*ESE 186, indicating that the ESER contains the binary value 10111010.

## \*ESR? (Query Only)

This is a query only command that returns the contents of the Standard Event Status Register (SESR). This query also clears the SESR, since reading the SESR clears it. For a more detailed discussion of the use of these registers, see Registers.

**Group** Status and Error

**Syntax** \*ESR?

**Related Commands** ALLEv?, \*CLS, DESE, \*ESE, EVENT?, EVMsg?, \*SRE, \*STB?

**Examples** \*ESR? might return the value 213, showing that the SESR contains binary 11010101.

## EVENT? (Query Only)

This is a query only command that returns an event code from the Event Queue that provides information about the results of the last \*ESR? read. EVENT? also removes the returned value from the Event Queue. For more information, see Event Handling for more information.

**Group** Status and Error



**Syntax** EVENT?

**Related Commands** ALLEv?, \*CLS, DESE, \*ESE, \*ESR?, EVMsg?, \*SRE, \*STB?

**Examples** EVENT? might return the response EVENT 110, indicating that there was an error in a command header.

## EVMsg? (Query Only)

This is a query only command that removes a single event code from the Event Queue that is associated with the results of the last \*ESR? read and returns the event code along with an explanatory message. For more information, see Event Handling for more information.

**Group** Status and Error

**Syntax** EVMsg?

**Related Commands** ALLEv?, \*CLS, DESE, \*ESE, \*ESR?, EVENT?, \*SRE, \*STB?

**Returns** The event code and message in the following format:

```
<Event Code><Comma><QString>[<Event Code><Comma>
<QString>...]<QString>::= <Message>;[<Command>]
```

where <Command> is the command that caused the error and may be returned when a command error is detected by the instrument. As much of the command will be returned as possible without exceeding the 60-character limit of the <Message> and <Command> strings combined. The command string is right justified.

**Examples** EVMSG? might return EVMSG 110, "COMMAND HEADER ERROR".

## EVQty? (Query Only)

This query only command returns the number of event codes that are in the Event Queue. This is useful when using the ALLEv? query, since it lets you know exactly how many events will be returned.

**Group** Status and Error

**Syntax** EVQty?

**Related Commands** ALLEv?, EVENT?, EVMsg?

**Examples** EVQTY? might return EVQTY 3, indicating that there are three event codes in the Event Queue.

## EXPort (No Query Form)

This command (no query form) exports waveform data as either a column of comma separated Y,T (vertical, horizontal) text pairs (.CSV) or as comma separated Y text values (.TXT). Histogram data is exported as comma separated text values only. Exporting data allows you to use it with other analysis tools. Exported waveforms cannot be imported. If you intend to recall a waveform later, you should save it instead of exporting it.

**Group** Save and Recall

**Syntax** EXPort { CH | MATH | REF }[, { MAIn | MAG1 | MAG2 }],  
<filepath>  
EXPort HIST <filepath>

- Arguments**
- CH<x> selects a channel waveform to export. The range for x is 1 through 8.
  - MATH<x> selects a math waveform to export. The range for x is 1 through 8.
  - REF<x> selects a reference waveform to export. The range for x is 1 through 8.
  - HIST selects histogram data to export.
  - MAIn (optional) selects Main as the timebase for the exported waveform. This is the default timebase.
  - MAG1 (optional) selects MAG1 as the timebase for the exported waveform.
  - MAG2 (optional) selects MAG2 as the timebase for the exported waveform.
  - <file path> is the location to which the waveform will be exported.

The <file path> is a quoted string that defines the file name and path. Input the file path using the form <drive>/<dir>/<filename>. The <filename> can be a Windows long file name. Do not use wild card characters.

**Examples** EXPORT REF2, "C:MY DOCUMENTS\DSA8300\MAINR2.TXT" exports the REF2 waveform data to the specified file.

## EXPort:TYPE

This command sets or queries the form of the exported waveform data. The data can be exported as either a column of comma separated Y,T (vertical, horizontal) text pairs (.CSV) or as comma separated Y text values (.TXT). This setting is only active when exporting waveform data.

**Group** Save and Recall

**Syntax** `EXPort:TYPE { NUMerictxt | SPReadsheetcsv }`

**Related Commands** [EXPort](#)

- Arguments**
- `NUMerictxt` selects the waveform data to be exported as a string of numeric text in ASCII form.
  - `SPReadsheetcsv` selects the waveform data to be exported as a column of comma separated X,Y pairs in ASCII form.

**Examples** `EXPORT:TYPE NUMERICTXT` sets the waveform data to be exported as ASCII text.  
`EXPORT:TYPE?` queries the setting for exporting waveform data.

## FACTory (No Query Form)

This command (no query form) resets the instrument to its factory default settings. Sending this command is equivalent to pressing the front-panel DEFAULT SETUP button or selecting Default Setup in the Utilities menu.

Executing the `FACTory` command does the following actions:

- Clears the Event Status Enable Register
- Clears the Service Request Enable Register
- Sets the Device Event Status Enable Register to 255
- Sets the Power On Status Clear Flag to TRUE
- Enables all Command Headers (HEADer ON)
- Clears the pending operation flag and associated operations
- Clears all references from the reference slots

The FACtory command does not change the following:

- The state of the GPIB (IEEE Std 488.1-1987) interface
- The selected GPIB address
- Data transfer settings
- Calibration data that affects device specifications
- Protected user data
- Stored settings

**Group** Miscellaneous

**Syntax** FACtory

**Related Commands** [RECALL:SETUp](#), [\\*RST](#), [SAVE:SETUp](#)

**Examples** FACTORY resets the instrument to its factory default settings.

## FILESystem:READFile? (Query Only)

Outputs the specified file to the GPIB port with maximum file size of 10 Mb

**Group** Miscellaneous

**Syntax** FILESystem:READFile?

**Arguments** FILE PATH is a quoted string that defines the file name and path. Input the file path using the form "<drive>\dir\filename". The <filename> can be a Windows long file name. Do not use wild card characters.

**Examples** FILESYSTEM:READFILE? "c:\My Documents\TekScope\PI\test.txt"

## HARDCopy (No Query Form)

This command (no query form) sends a copy of the screen display followed by an EOI to the hard copy port. Sending this command is equivalent to pressing the front-panel PRINT button.

---

**NOTE.** *This command is NOT IEEE STD 488.2-1987 compatible.*

---

**Group**     Hardcopy

**Syntax**     HARDCopy { START }

**Related Commands**     [HARDCopy:FORMat](#), [HARDCopy:FILEName](#), [HARDCopy:INKSaver](#)

**Arguments**     START initiates a screen copy that is sent to the controller where it is redirected to a printing device.

---

**NOTE.** *Use the \*WAI command between HARDCopy START commands to ensure that the first hardcopy task is sent to the printer before starting another.*

---

**Examples**     HARDCOPY START starts a hardcopy.

## HARDCopy:FILEName (No Query Form)

This command immediately sends a copy of the screen display to the file specified by the command argument. The filename may include a relative path, or a fully qualified path, to any location supported by Microsoft Windows, including the local hard disk, floppy drive, and network folders.

---

**NOTE.** *If the file you specify doesn't exist, sending this command creates it. If the file does exist, sending this command overwrites its contents.*

---

The image-file format that your hardcopy will have depends on whether you include a file extension when you specify the file path:

---

**NOTE.** *If your FILEName argument does not include a file extension, the file extension associated with the format set by the HARDCopy:FORMat command is automatically appended and that image-file format is used. You can change the file format using the HARDCopy:FORMat command, sending it before you send the HARDCopy:FILEName command.*

---

If your FILEName argument does include a file extension, the image-file associated with it is used instead. The file extension must be one associated with one of the formats listed for the HARDCopy:FORMat command.

<b>Group</b>	Hardcopy
<b>Syntax</b>	<code>HARDCopy:FILENAME &lt;file path&gt;</code>
<b>Related Commands</b>	<a href="#">HARDCopy</a> , <a href="#">HARDCopy:FORMAT</a> , <a href="#">HARDCopy:INKSaver</a>
<b>Arguments</b>	<p>&lt;FILE PATH&gt; specifies the file to which the hardcopy is sent. This argument is a quoted string that defines the file name, path, and, optionally, the file format if you include a file extension.</p> <hr/> <p><b>NOTE.</b> <i>The specified filename must be enclosed with quotation marks and wild card characters are illegal.</i></p> <hr/>
<b>Examples</b>	<p><code>HARDCOPY:FILENAME "C:\MY DOCUMENTS\TEK"</code> immediately outputs a hardcopy to the file TEK.&lt;format&gt;, where &lt;format&gt; is the image-file format set by the <code>HARDCopy:FORMAT</code> command, or by the optional file-name extension.</p> <p><code>HARDCOPY:FILENAME "C:\MY DOCUMENTS\TEK.PNG"</code> immediately outputs a hardcopy to the file TEK.PNG, where PNG is image-file format set by the file-name extension included in the command argument.</p>

## HARDCopy:FORMAT (No Query Form)

This command selects the image-file format used with the hardcopy commands. Several image file formats are provided to support compatibility with image-processing applications and efficient utilization of disk capacity.

<b>Group</b>	Hardcopy
<b>Syntax</b>	<code>HARDCopy:FORMAT { PNG   JPEG   BMP   TIFF   TARGA }</code>
<b>Related Commands</b>	<a href="#">HARDCopy:FILENAME</a> , <a href="#">HARDCopy</a>

- Arguments**
- PNG sets the hardcopy format to PNG. This setting produces compressed image files between 20KB and 100KB. This format uses a lossless image compression algorithm and will always produce the smallest file size of the available formats.
  - JPEG sets the hardcopy format to JPEG. This setting produces compressed image files between 60KB and 100KB. This format uses a lossy image compression algorithm.
  - BMP sets the hardcopy format to BMP. This setting produces minimally compressed (RLE) image files around 1MB. This is the default image file format and is commonly supported by Microsoft Windows compatible applications.
  - TIFF sets the hardcopy format to TIFF. This setting produces uncompressed image files that are typically larger than 1MB.
  - TARGA sets the hardcopy format to TARGA. This setting produces minimally compressed (RLE) image files around 1MB.
- Examples**    `HARDCOPY:FORMAT PNG` selects PNG as the file format to which hardcopy data is sent.

## HARDCopy:INKSaver (No Query Form)

This command turns Ink-saver mode on and off. When Ink-saver mode is on, hardcopies are produced with a white background to conserve ink. Ink-saver mode can be used when sending hardcopy images to a printer, or when sending hardcopy images to a file using the `HARDCopy:FILENAME` command.

**Group**    Hardcopy

**Syntax**    `HARDCopy:INKSaver { ON | OFF | NR1 }`

**Related Commands**    [HARDCopy](#), [HARDCopy:FORMat](#), [HARDCopy:FILEName](#)

- Arguments**
- ON enables the ink-saver mode.
  - OFF disables the ink-saver mode.
  - NR1 set to 0 disables the ink-saver mode; any other integer enables the Waveform readout display.

**Examples**    `HARDCOPY:INKSAVER ON` turns ink-saver mode on for subsequent hardcopy commands.

`HARDCOPY:INKSAVER?` returns `HARDCOPY:INKSAVER 1` if ink-saver mode is currently on.

## HEADer

This command sets or queries the Response Header Enable State that causes the instrument to either include or omit headers on query responses. This command does not affect IEEE Std 488.2-1987 Common Commands (those starting with an asterisk); these commands never return headers.

**Group** Miscellaneous

**Syntax** `HEADer { ON | OFF | NR1 }`  
`HEADer?`

**Related Commands** [VERBose](#)

- Arguments**
- `ON` sets the Response Header Enable State to true. This causes the instrument to include headers on applicable query responses. You can then use the query response as a command.
  - `OFF` sets the Response Header Enable State to false. This causes the instrument to omit headers on query responses so that only the argument is returned.
  - `NR1` set to 0 sets the Response Header Enable State to false, which disables the headers; any other integer sets the Response Header Enable State to true, which enables the headers. (A query always returns a 0 or 1.)

**Examples** `HEADER OFF` causes the instrument to omit headers from query responses.

`HEADER?` might return `HEADER 1`, indicating that the Response Header Enable State is on.

---

**NOTE.** *A query always returns a 0 or 1.*

---

## HIStogram? (Query Only)

This is a query only command that returns histogram parameters; it queries the state of all settable histogram parameters.

**Group** Histogram



**Syntax** HISTogram?

**Related Commands** [EXPort](#)

**Examples** HISTOGRAM? might return HISTOGRAM:BOX 1.5180000000E-006, -24.4980000000, 3.5180000000E-006, -74.4980000000;BOXPCNT 30.0000000000,25.0000000000, 7.0000000000,7.5000000000;COLOR 65535; ENABLE 1; DISPLAY 1; TYPE LINEAR; MODE VERTICAL; SIZE 4.0000000000; SOURCE CH3,MAIN; STATISTICS:MEAN 46.9321597622E-003; MEDIAN 224.0000000000E-003; STDDEV 227.027807252E-003; PKTOPK 500.0000000000E-003; SIGMA1 636.054733146E-003; SIGMA2 636.054733146E-003; SIGMA3 1.0000000000; PEAKHITS 39.4280000000E+003;HITS 185.7010000000E+003; WAVEFORMS 924.000000000; :HISTOGRAM:WFMDB:STATE 1.

## HISTogram:BOX

Defines the left, top, right, and bottom boundaries of the histogram box in source waveform coordinates. This is equivalent to setting the Limit Controls for Left, Top, Right, and Bottom in the Histogram Setup dialog box. This command is similar to the HISTogram:BOXPcnt command, which uses percentage coordinates to define the boundaries of the histogram box.

**Group** Histogram

**Syntax** HISTogram:BOX <NR3>,<NR3>,<NR3>,<NR3>  
HISTogram:BOX?

**Related Commands** [HISTogram:BOXPcnt](#)

- Arguments**
- NR3 (first) is the left position of the histogram box in source waveform coordinates.
  - NR3 (second) is the top position of the histogram box in source waveform coordinates.
  - NR3 (third) is the right position of the histogram box in source waveform coordinates.
  - NR3 (forth) is the bottom position of the histogram box in source waveform coordinates.

**Examples** HISTOGRAM:BOX 1.518E-006, -2.46E-1, 3.518E-6, -7.47E-1 defines the coordinates of the histogram box in source waveform coordinates.

HISTOGRAM:BOX? might return HISTOGRAM:BOX  
 1.51800000000E-006, -0.246000000000,  
 3.51800000000E-006, -0.747000000000.

## HISStogram:BOXPcnt

Defines the left, top, right, and bottom positions of the histogram box, in percentage of screen coordinates. The upper left has the value 0, 0 and the lower right has the value 100, 100. Any value outside the range of 0 to 100 is clipped and generates an error. This command is similar to the HISStogram:BOX command, which uses waveform coordinates to define the boundaries of the histogram box.

**Group** Histogram

**Syntax** HISStogram:BOXPcnt <NR3>, <NR3>, <NR3>, <NR3>  
 HISStogram:BOXPcnt?

**Related Commands** [HISStogram:BOX](#)

- Arguments**
- NR3 (first) is the left position of the histogram box in percentage coordinates. The default value is 30.
  - NR3 (second) is the top position of the histogram box in percentage coordinates. The default value is 25.
  - NR3 (third) is the right position of the histogram box in percentage coordinates. The default value is 70.
  - NR3 (forth) is the bottom position of the histogram box in percentage coordinates. The default value is 75.

**Examples** HISTOGRAM:BOXPCNT 35, 21, 72, 76 defines the coordinates of a histogram box in percentage coordinates.

HISTOGRAM:BOXPCNT? might return HISTOGRAM:BOXPCNT 30.0000000000,  
 25.0000000000, 70.0000000000, 75.0000000000.

## HISStogram:COLOr

This command sets or queries the color of the histogram data and its boundary area. (This command is the same as the DISPlay:COLOr:HISStogram command.)

This is equivalent to setting Color in the Display Options section of the Histogram Setup dialog box. The default histogram color is aqua.

The argument for the command is an integer equal to the sum of three color values (red, green, and blue). You specify the value for each of the three color values and sum them to specify the background color.

The following table lists values for common colors.

**Index number values for common colors**

Name	Index number
Black	0
Blue	16711680
Aqua	16776960
Lime	65280
Fuchsia	16711935
Red	255
Yellow	65535
White	16777215
Navy	8388608
Teal	8421376
Green	32768
Purple	8388736
Maroon	128
Olive	32896
Gray	8421504
Silver	12632193

**Group** Histogram

**Syntax** HISTogram:COLor <NR1>  
 HISTogram:COLor?

**Related Commands** [DISplay:COLor:HIStogram](#)

**Arguments** NR1 is an integer in the range of 0 through 16777215 representing a color defined as follows:  $\text{red} + 256 \times \text{green} + 65536 \times \text{blue}$  where red, green and blue are values between 0 and 255.

**Examples** HISTOGRAM:COLOR 128 sets the color of the histogram data and its boundary area to maroon.

HISTOGRAM:COLOR? might return HISTOGRAM:COLOR:8421504, indicating that the color of the histogram data and its boundary is gray.

## HIStogram:COUNT (No Query Form)

This command clears the count and statistics for the histogram and the histogram source data. If the histogram is on, then counting restarts. This is equivalent to clicking Clear in the Source section of the Histogram Setup dialog box. There is no query form of this command.

**Group** Histogram

**Syntax** HIStogram:COUNT {RESET}

**Examples** HISTOGRAM:COUNT RESET clears the count and statistics for the histogram and the histogram source data.

## HIStogram:DISplay

This command sets or queries whether the histogram is displayed. This is equivalent to selecting Histogram in the Display Options section of the Histogram Setup dialog box.

**Group** Histogram

**Syntax** HIStogram:DISplay { ON | OFF | NR1 }  
HIStogram:DISplay?

- Arguments**
- ON displays the histogram.
  - OFF disables the histogram display.
  - NR1 set to 0 disables the histogram display; any other value displays the histogram.

**Examples** HISTOGRAM:DISPLAY ON displays the histogram.

HISTOGRAM:DISPLAY? might return HISTOGRAM:DISPLAY 0, indicating that the histogram display is disabled.

---

**NOTE.** A query always returns a 0 or 1.

---

## HIStogram:ENABle

This command sets or queries whether the histogram calculations are enabled. This is equivalent to selecting Enable Histogram in the Source section of the Histogram Setup dialog box. Sending this command will cause the histogram statistics to be displayed (but not the histogram). Use the HIStogram:DISplay command to control whether the histogram is displayed on screen.

**Group** Histogram

**Syntax** HIStogram:ENABle { ON | OFF | NR1 }  
HIStogram:ENABle?

- Arguments**
- ON enables histogram calculations.
  - OFF disables histogram calculations.
  - NR1 set to 0 disables the histogram calculations; any other value enables histogram calculations.

**Examples** HISTOGRAM:ENABLE ON enables histogram calculations.  
HISTOGRAM:ENABLE? might return HISTOGRAM:ENABLE 0, indicating that the histogram calculations are disabled.

---

**NOTE.** A query always returns a 0 or 1.

---

## HIStogram:MODE

Selects the type of histogram to create. This is equivalent to selecting either Vertical or Horizontal in the Source section of the Histogram Setup dialog box.

**Group** Histogram

**Syntax** HIStogram:MODE {HORizontal | VERTical}  
HIStogram:MODE?

- Arguments**
- HORizontal displays a horizontally positioned histogram that shows time distribution.
  - VERTical displays a vertically positioned histogram that shows a voltage distribution (or other vertical distribution, such as amperes).

- Examples** HISTOGRAM:MODE HORIZONTAL sets the histogram mode to the horizontal display mode.
- HISTOGRAM:MODE? might return HISTOGRAM:MODE VERTICAL, indicating that the vertical display mode is selected for the histogram.

## HIStogram:SIZE

This command sets or queries the width or height in divisions of the histogram plot on the screen. This is equivalent to setting Size in the Display Options section of the Histogram Setup dialog box.

- Group** Histogram
- Syntax** HIStogram:SIZE <NR3>  
HIStogram:SIZE?
- Arguments** NR3. The value can vary from 1.0 to 10.0 in Horizontal mode and from 1.0 to 10.0 in Vertical mode. Resolution is to the nearest pixel.
- Examples** HISTOGRAM:SIZE 2 sets the size of the histogram to 2 divisions.
- HISTOGRAM:SIZE? might return HISTOGRAM:SIZE 4.0000000000, indicating the histogram size is set to 4 divisions.

## HIStogram:SOURce

This command sets or queries which source will be compared against the histogram box when the histogram testing is enabled. You can also select the timebase on which to perform testing. If you do not specify a timebase, Main is the default. This is equivalent to selecting the source from the pulldown list in the Source section of the Histogram Setup dialog box. The waveform need not be displayed for histograms to run. You might want the channel displays disabled so you can see a full-screen histogram and not have waveform data confuse the display.

- Group** Histogram
- Syntax** HIStogram:SOURce {CH<x> | MATH<x> | REF<x>} [, {MAIn | MAG1 | MAG2}]  
HIStogram:SOURce?

- Arguments**
- CH<x> selects a channel waveform as the source for the histogram. The range for x is 1 through 8.
  - MATH<x> selects a math waveform as the source for the histogram. The range for x is 1 through 8.
  - REF<x> selects a reference waveform as the source for the histogram. The range for x is 1 through 8.
  - MAIn (optional) selects Main as the timebase for the histogram display.
  - MAG1 (optional) selects MAG1 as the timebase for the histogram display.
  - MAG2 (optional) selects MAG2 as the timebase for the histogram display.

**Examples** HISTOGRAM:SOURCE CH1 enables the Channel 1 waveform to be compared against the histogram box. The default timebase, Main, is used for the histogram display.

HISTOGRAM:SOURCE? might return HISTOGRAM:SOURCE CH1,MAG1, indicating that the waveform for Channel 1 is the source for the histogram and that it is displayed using MAG1 as the timebase.

## HISTogram:STATistics? (Query Only)

This query only command returns all histogram statistics. These are the Histogram statistics that are displayed in the Histogram readout:

- Mean
- Median
- Std Dev
- Pk-Pk
- $\mu \pm 1s$
- $\mu \pm 2s$
- $\mu \pm 3s$
- Peak
- Hits
- Waveforms

The histogram statistics (and count) can be reset with the HISTogram:COUNT command.

**Group** Histogram

**Syntax** HISTogram:STATistics?

**Related Commands** [HISTogram:COUNt](#)

**Examples** HISTOGRAM: STATISTICS? might return the following:  
HISTOGRAM: STATISTICS: MEAN 43.000000000E-009; MEDIAN  
43.000000000E-009; STDDEV 5.80230431544E-009; PKTOPK  
20.000000000E-009; SIGMA1 58.7064; SIGMA2 58.7064; SIGMA3  
100.0000; PEAKHITS 2.4380000000E+003; HITS  
490.038000000E+003; WAVEFORMS 2.4380000000E+003

## HISTogram:STATistics:HITS? (Query Only)

This query only command returns the total hits calculated for the histogram. This is the value that is displayed for Hits in the Histogram readout.

**Group** Histogram

**Syntax** HISTogram:STATistics:HITS?

**Examples** HISTOGRAM: STATISTICS: HITS? might return  
HISTOGRAM: STATISTICS: HITS 6.8340000000E+003, indicating that the  
total hits for the histogram is 6,834.

## HISTogram:STATistics:MEAN? (Query Only)

This query only command returns the mean value calculated for the histogram. This is the value that is displayed for Mean in the Histogram readout.

**Group** Histogram

**Syntax** HISTogram:STATistics:MEAN?

**Examples** HISTOGRAM: STATISTICS: MEAN? might return  
HISTOGRAM: STATISTICS: MEAN 43.000000000E-009, indicating that the  
mean value for the horizontal histogram is 43 ns.



## HISTogram:STATistics:MEDIAN? (Query Only)

This query only command returns the median value calculated for the histogram. This is the value that is displayed for Median in the Histogram readout.

**Group** Histogram

**Syntax** HISTogram:STATistics:MEDIAN?

**Examples** HISTOGRAM:STATISTICS:MEDIAN? might return  
HISTOGRAM:STATISTICS:MEDIAN 43.0000000000E-009, indicating that the median value for the horizontal histogram is 43 ns.

## HISTogram:STATistics:PEAKHits? (Query Only)

This query only command returns the peak hits calculated for the histogram. This is the value that is displayed for Peak in the Histogram readout.

**Group** Histogram

**Syntax** HISTogram:STATistics:PEAKHits?

**Examples** HISTOGRAM:STATISTICS:PEAKHITS? might return  
HISTOGRAM:STATISTICS:PEAKHITS 370.000000000, indicating that the peak hits value calculated for the waveform is 370.

## HISTogram:STATistics:PKTOPK? (Query Only)

This query only command returns the peak-to-peak value calculated for the histogram. This is the value that is displayed for Pk-Pk in the Histogram readout.

**Group** Histogram

**Syntax** HISTogram:STATistics:PKTOPK?

**Examples** HISTOGRAM:STATISTICS:PKTOPK? might return  
HISTOGRAM:STATISTICS:PKTOPK 20.0000000000E-009, indicating that the peak-to-peak value for the horizontal histogram is 20 ns.

## HISTogram:STATistics:SIGMA<x>? (Query Only)

This query only command returns the population densities for  $\pm 1s$ ,  $\pm 2s$ , or  $\pm 3s$  value (standard deviation from the mean value) calculated for the histogram. This is the value that is displayed for  $\mu \pm 1s$ ,  $\mu \pm 2s$ , or  $\mu \pm 3s$  in the Histogram readout.

**Group** Histogram

**Syntax** HISTogram:STATistics:SIGMA<x>?

**Arguments** <x> (1, 2, or 3) is the value indicating a query for the sigma1, sigma2, or sigma3 value.

**Examples** HISTOGRAM:STATISTICS:SIGMA1? might return  
 HISTOGRAM:STATISTICS:SIGMA1 58.0060, indicating that 58% of the hits fall within one standard deviation ( $\mu \pm 1s$ ) from the mean calculated for the histogram.

## HISTogram:STATistics:STDdev? (Query Only)

This query only command returns the standard deviation value calculated for the histogram. This is the value that is displayed for Std Dev in the Histogram readout.

**Group** Histogram

**Syntax** HISTogram:STATistics:STDdev?

**Examples** HISTOGRAM:STATISTICS:STDDEV? might return  
 HISTOGRAM:STATISTICS:STDDEV 5.80230767128E-009, indicating that the standard deviation value for the horizontal histogram is 5.80 ns.

## HISTogram:STATistics:WAVEforms? (Query Only)

This query only command returns the number of waveforms used in the histogram. This is the value that is displayed for Waveforms in the Histogram readout.

**Group** Histogram

**Syntax** HISTogram:STATistics:WAVEforms?

**Examples** HISTOGRAM:STATISTICS:WAVEFORMS? might return  
 HISTOGRAM:STATISTICS:WAVEFORMS 2.0810000000E+003, indicating that  
 2081 waveforms were used to create the histogram.

## HIStogram:TYPE

This command sets or queries whether the histogram is displayed linearly or logarithmically. This is equivalent to selecting Linear or Logarithmic in the Display Options section of the Histogram Setup dialog box.

**Group** Histogram

**Syntax** HISTogram:TYPE {LINEAR | LOG}  
 HISTogram:TYPE?

- Arguments**
- LINEAR specifies that bin counts smaller than the maximum should be scaled linearly by dividing the bin count by the maximum bin count.
  - LOG specifies that bin counts smaller than the maximum should be scaled logarithmically ( $\log(\text{bin-count})$ ) with  $\log(0)$  staying at 0 (baseline). The base of the log does not matter since logs to different bases differ only by a constant multiplier. Logarithmic scaling provides better visual detail for bins with low counts.

**Examples** HISTOGRAM:TYPE LINEAR displays the count in each bin scaled linearly.  
 HISTOGRAM:TYPE? might return HISTOGRAM:TYPE LINEAR, indicating that the histogram display is scaled linearly.

## HIStogram:WFMDB:STATE

This command sets or queries whether the histogram counting is made on a waveform database created on the source of the histogram. This is equivalent to selecting or clearing the Use Wfm Database check box in the Source section of the Histogram Setup dialog box.

**Group** Histogram

**Syntax** HISTogram:WFMDB:STATE { ON | OFF | NR1 }  
 HISTogram:WFMDB:STATE?

- Arguments**
- ON enables histogram counting on a waveform database created on the source of the histogram.
  - OFF disables histogram counting on a waveform database; testing is done “on the fly.”
  - NR1 set to 0 disables the histogram counting on a waveform database; any other value enables histogram counting on a waveform database.

**Examples** HISTOGRAM:WFMDB:STATE ON enables histogram counting on a waveform database.

HISTOGRAM:WFMDB:STATE? might return HISTOGRAM:WFMDB:STATE 0, indicating that the histogram counting on a waveform database is disabled.

---

**NOTE.** *A query always returns 0 or 1.*

---

## HORizontal? (Query Only)

This query only command returns all settings for the horizontal commands.

**Group** Horizontal

**Syntax** HORizontal?

**Examples** HORIZONTAL? might return the following string (partial):

```
HORIZONTAL:UNITS S; BITS:STANDARD OC192; BITRATE
9.95328000000E+009; HORIZONTAL:EXT10MHZREF:FREQ
10.0000000000E+006; HORIZONTAL:MAIN:SCALE
5.0000000000E-009; POSITION 81.1630000000.009;
RESOLUTION 12.5000000000E-012; RECORDLENGTH 4000;REFPOINT
1.0000000000; TOLPOINT 81.1630000000E-009; TOFPOINT
31.1630000000E-009; HORIZONTAL:MAGNIFY1:SCALE
2.0000000000E-009; POSITION 51.1630000000E-009;
RESOLUTION 10.0000000000E-012; RECORDLENGTH 2000;REFPOINT
1.0000000000; VIEW 1; TOLPOINT 51.1630000000E-009;
TOFPOINT 31.1630000000E-009; HORIZONTAL:MAGNIFY2:SCALE
3.3000000000E-011; POSITION 31.2130000000E-009;
RESOLUTION 220.0000000000E-015; RECORDLENGTH 1000;REFPOINT
150.0000000000E-003; VIEW 0;TOLPOINT 31.4930000000E-009;
TOFPOINT 31.1630000000E-009; ...
```

## HORizontal:BITS:BITRate

This command sets or queries the bit rate of the timebase. Sending this command is equivalent to setting a value in the Bit Rate control in the All Timebases section of the Horizontal Setup dialog box. This value is used to convert horizontal scales from seconds to bits.

If a communications standard is set by the HORizontal:BITS:STANdard command, the bit rate is coerced to the bit rate for the specified standard.

**Group** Horizontal

**Syntax** HORizontal:BITS:BITRate <NR3>  
HORizontal:BITS:BITRate?

**Related Commands** [HORizontal:BITS:STANdard](#), [HORizontal:UNIts](#)

**Arguments** NR3 is the bit rate.

**Examples** HORizontal:BITS:BITRate 5.0E6 sets the bit rate to 5 Mbits per second.  
HORizontal:BITS:BITRate? might return HORizontal:BITS:BITRate 2.48832000000E+009, indicating that the bit rate is set to 2.488 Gbits per second.

## HORizontal:BITS:STANdard

This command sets or queries the communication standard, which sets the bit rate used to calculate parameters, such as horizontal position, when HORizontal:UNIts are set to the BITS selection. The seconds-to-bits conversion factors of all timebases are set in accordance with the bit rate associated with the selected communication standard. Sending this command is equivalent to selecting a communication standard for the Comm Standard control in the All Timebases section of the Horizontal Setup dialog box.

---

**NOTE.** Setting the HORizontal:BITS:STANdard with this command sets the MASK:STANdard setting to the same argument. Also, setting the horizontal bit rate with the HORizontal:BITS:BITRate command coerces the HORizontal:BITS:STANdard setting to USERMask.

---

**Group** Horizontal

**Syntax**    `HORIZONTAL:BITS:STANDARD { NONE | ATARXG1 | ATARXG2 | ATARXG3 | ATATXG1 | ATATXG2 | ATATXG3 | ENET40GB_LR4 | ENET40GB_SR4 | ENET100GB_ER4 | ENET100GB_LR4 | ENET100GB_SR10 | ENET1250 | ENET2500 | ENET3125 | ENET9953 | ENET10313 | ENET11096 | ENET41250 | FC133 | FC133E | FC266 | FC266E | FC531 | FC531E | FC1063 | FC1063E | FC2125 | FC2125E_ABR | FC2125E_ABT | FC2125E_AGR | FC2125E_AGT | FC4250E_ABR | FC4250E_ABT | FC4250E_AGR | FC4250E_AGT | FC8500D | FC8500E_ABR | FC8500E_ABT | FC8500E_AGR | FC8500E_AGT | FC8500FINAL | FC4250 | FC10519 | FC11317 | FC14025_MMR6_1 | FC14025_SMR6_1 | FEC2666 | FEC10664 | FEC10709 | FEC42657 | FEC43018 | INF2500 | INFIE25 | INFIniband | OC1 | OC3 | OC9 | OC12 | OC18 | OC24 | OC36 | OC48 | OC192 | OC768 | PCIEXPRESS_RCV | PCIEXPRESS50_RCV | RIO_SERIAL1G | RIO_SERIAL2G | RIO_SERIAL3G | SAS3_0_XR | SAS3_0_XR_AASJ | SAS3_0_SATA | USERMask | XFI9950_TAA | XFI9950_RAD | XFI9950_THB | XFI9950_RHC | XFI9950_TMBP | XFI9950_RMCP | XAUIFar | XAUINear | XAUINear}`  
**HORIZONTAL:BITS:STANDARD?**

- Arguments**
- NONE disables the communications standards.
  - ATARXG1 selects the Serial ATA, G1 Rx, 1.5 Gb/s bit rate and mask.
  - ATARXG2 selects the Serial ATA, G2 Rx, 3 Gb/s bit rate and mask.
  - ATARXG3 selects the Serial ATA, G2 Rx, 6 Gb/s bit rate and mask.
  - ATATXG1 selects the Serial ATA, G1 Tx, 1.5 Gb/s bit rate and mask.
  - ATATXG2 selects the Serial ATA, G2 Tx, 3 Gb/s bit rate and mask.
  - ATATXG3 selects the Serial ATA, G2 Tx, 6 Gb/s bit rate and mask.
  - ENET40GB\_LR4 selects the 10.3125 Gb/s bit rate and mask, in accordance with the Ethernet 40GBASE-LR4 standard.
  - ENET40GB\_SR4 selects the 10.3125 Gb/s bit rate and mask, in accordance with the Ethernet 40GBASE-SR4 standard.
  - ENET100GB\_ER4 selects the 27.78125 Gb/s bit rate and mask, in accordance with the Ethernet 100GBASE-ER4 standard.
  - ENET100GB\_LR4 selects the 27.78125 Gb/s bit rate and mask, in accordance with the Ethernet 100GBASE-LR4 standard.
  - ENET100GB\_SR10 selects the 10.3125 Gb/s bit rate and mask, in accordance with the Ethernet 100GBASE-SR10 standard.
  - ENET1250 selects the 1.250 Gb/s bit rate and mask, in accordance with the IEEE 802.3 standard.
  - ENET2500 selects the 2.500 Gb/s bit rate (2x Gigabit Ethernet) and mask.

- ENET3125 selects the 3.125 Gb/s bit rate and mask, in accordance with the IEEE 802.3ae-2002 standard.
- ENET9953 selects the 9.95328 Gb/s bit rate and mask, in accordance with the IEEE 802.3ae-2002 standard.
- ENET10313 selects the 10.3125 Gb/s bit rate and mask, in accordance with the IEEE 802.3ae-2002 standard.
- ENET11096 selects the 11.095728 Gb/s bit rate, which is a forward error-correction rate for 10GbE (10.3125 Gb/s per IEEE 802.3ae-2002), and mask.
- ENET41250 selects the 41.2500 Gb/s bit rate and mask in accordance with the Ethernet 40GBASE-FR standard.
- FC133 selects the 132.81 Mb/s bit rate and mask, in accordance with the ANSI X3.230-1994 standard.
- FC133E selects the 132.81 Mb/s bit rate and mask, in accordance with the ANSI X3.230-1994 standard.
- FC266 selects the 265.6 Mb/s bit rate and mask, in accordance with the ANSI X3.230-1994 standard.
- FC266E selects the 265.6 Mb/s bit rate and mask, in accordance with the ANSI X3.230-1994 standard.
- FC531 selects the 531.2 Mb/s bit rate and mask, in accordance with the ANSI X3.230-1994 standard.
- FC531E selects the 531.2 Mb/s bit rate and mask, in accordance with the ANSI X3.230-1994 standard.
- FC1063 selects the 1.0625 Gb/s bit rate and mask, in accordance with the ANSI X3.230-1994 standard.
- FC1063E selects the 1.0625 Gb/s bit rate and mask, in accordance with the ANSI X3.230-1994 standard.
- FC2125 selects the 2.125 Gb/s bit rate and mask, in accordance with the ANSI Fibre Channel Physical Interface (FC-PI) Rev 11 draft standard.
- FC2125E\_ABR selects the 2.125 Gb/s bit rate and mask, in accordance with ANSI X3.230-1999 NCITS 1235D/Rev 11, Abs, Beta, Recv.
- FC2125E\_ABT selects the 2.125 Gb/s bit rate and mask, in accordance with ANSI X3.230-1999 NCITS 1235D/Rev 11, Abs, Beta, Transm.
- FC2125E\_AGR selects the 2.125 Gb/s bit rate and mask, in accordance with ANSI X3.230-1999 NCITS 1235D/Rev 11, Abs, Gamma, Recv.
- FC2125E\_AGT selects the 2.125 Gb/s bit rate and mask, in accordance with ANSI X3.230-1999 NCITS 1235D/Rev 11, Abs, Gamma, Transm.

- FC4250 selects the 4.250 Gb/s bit rate and mask, in accordance with the ANSI Fibre Channel Physical Interface (FC-PI) Rev 11 draft standard.
- FC4250E\_ABR selects the 4.25 Gb/s bit rate and mask, in accordance with FIBRE CHANNEL PHYSICAL INTERFACES (FC-PI-2), Rev 3.0 9/13/02 Abs, Beta, Recv.
- FC4250E\_ABT selects the 4.25 Gb/s bit rate and mask, in accordance with FIBRE CHANNEL PHYSICAL INTERFACES (FC-PI-2), Rev 3.0 9/13/02 Abs, Beta, Transm.
- FC4250E\_AGR selects the 4.25 Gb/s bit rate and mask, in accordance with FIBRE CHANNEL PHYSICAL INTERFACES (FC-PI-2), Rev 3.0 9/13/02, Abs, Gamma, Recv.
- FC4250E\_AGT selects the 4.25 Gb/s bit rate and mask, in accordance with FIBRE CHANNEL PHYSICAL INTERFACES (FC-PI-2), Rev 3.0 9/13/02, Abs, Gamma, Transm.
- FC8500D selects the draft 8.500 Gb/s optical bit rate and mask per T11 January 2006 proposal.
- FC8500E\_ABR selects the 8.500 Gb/s bit rate and mask, in accordance with FIBRE CHANNEL PHYSICAL INTERFACES (FC-PI-2), Rev 3.0 9/13/02 Abs, Beta, Recv.
- FC8500E\_ABT selects the 8.500 Gb/s bit rate and mask, in accordance with FIBRE CHANNEL PHYSICAL INTERFACES (FC-PI-2), Rev 3.0 9/13/02, Abs, Beta, Transm.
- FC8500E\_AGR selects the 8.500 Gb/s bit rate and mask, in accordance with FIBRE CHANNEL PHYSICAL INTERFACES (FC-PI-2), Rev 3.0 9/13/02, Abs, Gamma, Recv.
- FC8500E\_AGT selects the 8.500 Gb/s bit rate and mask, in accordance with FIBRE CHANNEL PHYSICAL INTERFACES (FC-PI-2), Rev 3.0 9/13/02, Abs, Gamma, Transm.
- FC8500FINAL selects the 8.500 Gb/s bit rate and mask, in accordance with the Fibre Channel Optical 8GFC standard.
- FEC2666 selects the 2.6660571 Gb/s bit rate and mask, in accordance with the ITU-T G.975 Recommendation.
- FC10519 selects the 10.51875 Gb/s bit rate and mask, in accordance with the proposed 10 Gigabit Fibre Channel (10GFC) standard.
- FC11317 selects the 11.317 Gb/s bit rate and mask, in accordance with the Fibre Channel Optical FC FEC standard.
- FC14025\_MMR6\_1 selects the 14.0250 Gb/s bit rate and mask, in accordance with the Fibre Channel Optical 16GFC MM r6.1 standard.



- FC14025\_SMR6\_1 selects the 14.0250 Gb/s bit rate and mask, in accordance with the Fibre Channel Optical 16GFC SM r6.1 standard.
- FEC10664 selects the 10.6642 Gb/s bit rate and mask, in accordance with the ITU-T G.975 Recommendation.
- FEC10709 selects the 10.709225 Gb/s bit rate and mask, in accordance with the ITU-T G.709 Draft Recommendation.
- FEC42657 selects the 42.6569 Gb/s bit rate and mask, in accordance with the ITU-T G.975 Recommendation.
- FEC43018 selects the 43.018414 Gb/s bit rate and mask, in accordance with the ITU-T G.709 Draft Recommendation.
- INF2500 selects the 2.500 Gb/s bit rate and mask, in accordance with the InfiniBand standard. Replaces the INFINIBAND argument. If INFINIBAND is set and a query is sent, the instrument returns INF2500.
- INFIE25 selects the 2.5 Gb/s bit rate and mask, in accordance with the InfiniBand, IBTA Spec 1.0a, 2.5 Electrical, 2.5 Gb/s standard.
- INFIniband selects the 2.500 Gb/s bit rate and mask, in accordance with the InfiniBand standard.
- OC1 selects the 51.84 Mb/s bit rate and mask, in accordance with the Sonet/SDH OC-1/STM-0 standard.
- OC3 selects the 155.52 Mb/s bit rate and mask, in accordance with the Sonet/SDH OC-3/STM-1 standard.
- OC9 selects the 466.56 Mb/s bit rate and mask, in accordance with the Sonet/SDH OC-9/STM-3 standard.
- OC12 selects the 622.08 Mb/s bit rate and mask, in accordance with the Sonet/SDH OC-12/STM-4 standard.
- OC18 selects the 933.12 Mb/s bit rate and mask, in accordance with the Sonet/SDH OC-18/STM-6 standard.
- OC24 selects the 1.24416 Gb/s bit rate and mask, in accordance with the Sonet/SDH OC-24/STM-8 standard.
- OC36 selects the 1.86624 Gb/s bit rate and mask, in accordance with the Sonet/SDH OC-36/STM-12 standard.
- OC48 selects the 2.488.32 Gb/s bit rate and mask, in accordance with the Sonet/SDH OC-48/STM-16 standard.
- OC192 selects the 9.95328 Gb/s bit rate and mask, in accordance with the Sonet/SDH OC192/STM-64 standard.
- OC768 selects the 39.8131 Gb/s bit rate and mask, in accordance with the Sonet/SDH OC-768/STM-256 standard.

- **PCIEXPRESS\_Rcv** selects the 2.5 Gb/s bit rate and mask, in accordance with the PCI Express Receiver, 2.5 Gb/s standard.
- **PCIEXPRESS50\_Rcv** selects the 5 Gb/s bit rate and mask, in accordance with the PCI Express Receiver, 5 Gb/s standard.
- **RIO\_SERIAL1G** selects the 1.25 Gb/s bit rate and mask, in accordance with the Rapid IO Serial, 1.25 Gb/s standard.
- **RIO\_SERIAL2G** selects the 2.5 Gb/s bit rate and mask, in accordance with the Rapid IO Serial, 2.5 Gb/s standard.
- **RIO\_SERIAL3G** selects the 3.25 Gb/s bit rate and mask, in accordance with the Rapid IO Serial, 3.25 Gb/s standard.
- **SAS3\_0\_XR** selects the 3 Gb/s bit rate and mask.
- **SAS3\_0\_XR\_AASJ** selects the 3 Gb/s applied sinusoidal jitter bit rate and mask.
- **SAS3\_0\_SATA** selects the 3 Gb/s bit rate and mask.
- **USERMask** selects user-defined masks for testing (see Description, above).
- **XFI9950\_TAA** selects the 9.95 Gb/s bit rate and mask.
- **XFI9950\_RAD** selects the 9.95 Gb/s bit rate and mask.
- **XFI9950\_THB** selects the 9.95 Gb/s bit rate and mask.
- **XFI9950\_RHC** selects the 9.95 Gb/s bit rate and mask.
- **XFI9950\_TMBP** selects the 9.95 Gb/s bit rate and mask.
- **XFI9950\_RMCP** selects the 9.95 Gb/s bit rate and mask.
- **XAUIFar** selects the 3.1250 Gb/s far-end normalized bit rate and mask, in accordance with the IEEE 802.3ae-2002 standard.
- **XAUIRFar** selects the 3.1250 Gb/s far-end relative bit rate and mask, in accordance with the IEEE 802.3ae-2002 standard.
- **XAUINear** selects the 3.1250 Gb/s near-end normalized bit rate and mask, in accordance with the IEEE 802.3ae-2002 standard.
- **XAUIRNear** selects the 3.1250 Gb/s near-end relative bit rate and mask, in accordance with the IEEE 802.3ae-2002 standard.

## **HORizontal:DISPlayscale:BITS**

This command sets or queries the display scale mode when horizontal units are set to BITS.

**Group**    Horizontal

**Syntax** `HORizontal:DISPlayscale:BITS {PERDivision | PERScreen}`  
`HORizontal:DISPlayscale:BITS?`

**Related Commands** [HORizontal:UNIts](#)

**Arguments** `PERDivision` sets the horizontal scale controls and readouts to display the time span for each division of the screen. This is the legacy behavior.

`PERScreen` sets the horizontal scale controls and readouts to display the time span of the entire graticule (10 divisions). This is the default setting for this command.

**Examples** `HORIZONTAL:DISPLAYSCALE:BITS PERSCREEN` sets the horizontal scale for all timebases to the time span for the entire screen (10 divisions) when the horizontal units is set to BITS.

`HORIZONTAL:DISPLAYSCALE:BITS?` might return `PERDivision`, which indicates that the instrument horizontal scale controls and readouts display the time span for each division of the screen.

## HORizontal:DISPlayscale:DISTance

This commands sets or queries the display scale mode when horizontal units are set to distance (meters, feet or inches).

**Group** Horizontal

**Syntax** `HORizontal:DISPlayscale:DISTance {PERDivision | PERScreen}`  
`HORizontal:DISPlayscale:DISTance?`

**Related Commands** [HORizontal:UNIts](#)

**Arguments** `PERDivision` sets the horizontal scale controls and readouts to display the time span for each division of the screen. This is the legacy behavior. This is the default setting for this command.

`PERScreen` sets the horizontal scale controls and readouts to display the time span of the entire graticule (10 divisions).

**Examples** `HORIZONTAL:DISPLAYSCALE:DISTANCE PERSCREEN` sets the horizontal scale for all timebases to the time span for the entire screen (10 divisions) when the horizontal units are distance when the horizontal units is set to DISTANCE.

`HORIZONTAL:DISPLAYSCALE:DISTANCE?` might return `HORIZONTAL:DISPLAYSCALE_DISTANCE PERDIVISION`, indicating that the display scale for all timebases is per division when the horizontal units is set to `DISTANCE`.

## HORizontal:DISPlayscale:SEConds

This commands sets or queries the display scale mode when horizontal units are set to `SEConds`.

**Group** Horizontal

**Syntax** `HORizontal:DISPlayscale:SEConds {PERDivision | PERScreen}`  
`HORizontal:DISPlayscale:SEConds?`

**Related Commands** [HORizontal:UNIts](#)

**Arguments** `PERDivision` sets the horizontal scale controls and readouts to display the time span for each division of the screen. This is the legacy behavior. This is the default setting for this command.

`PERScreen` sets the horizontal scale controls and readouts to display the time span of the entire graticule (10 divisions).

**Examples** `HORIZONTAL:DISPLAYSCALE:SECONDS PERSCREEN` sets the horizontal scale for all timebases to the time span for the entire screen (10 divisions) when the horizontal units are distance when the horizontal units is set to `SECONDS`.

`HORIZONTAL:DISPLAYSCALE:SECONDS?` might return `HORIZONTAL:DISPLAYSCALE_SECONDS PERDIVISION`, indicating that the display scale for all timebases is per division when the horizontal units is set to `SECONDS`.

## HORizontal:DISTance:DIElectric

This command sets or queries the dielectric constant. This is equivalent to setting the Dielectric constant in the Distance section of the Horizontal Setup dialog box.

The dielectric constant is a property of materials that can be used to compute the propagation velocity of signals through a particular medium. The propagation velocity is used to convert time to distance when the horizontal units are specified as distance. Propagation velocity ( $V_p$ ) is related to the dielectric constant ( $\epsilon$ ) by the following equation:

$$V_p = \frac{c}{e^{1/2}}$$

where c is the speed of light in a vacuum.

Setting the dielectric constant will cause the propagation velocity to be set as appropriate.

**Group** Horizontal

**Syntax** `HORizontal:DISTance:DIElectric <NR3>`  
`HORizontal:DISTance:DIElectric?`

**Related Commands** [HORizontal:UNIts](#), [HORizontal:DISTance:PVELocity](#)

**Arguments** NR3 is the dielectric constant of the material being tested. The range of values is 1 to 10,000. Typical values for PC board traces and cables range from approximately 1.5 to 10.

**Examples** `HORizontal:DISTance:DIElectric 3.3` sets the dielectric constant to 3.3. The propagation velocity (`HORizontal:DISTance:PVELocity`) is also set to the corresponding value.

`HORizontal:DISTance:DIElectric?` might return  
`HORizontal:DISTance_DIElectric 4.40000000`, indicating that the dielectric constant value is set to 4.40000.

## HORizontal:DISTance:PVELocity

This command sets or queries the Propagation Velocity. This is equivalent to setting the Propagation Velocity in the Distance section of the Horizontal Setup dialog box.

The propagation velocity is used to convert time to distance when the horizontal units are specified as distance. The dielectric constant (e) is related to propagation velocity (Vp) by the following equation:

$$e = \left(\frac{c}{V_p}\right)^2$$

where c is the speed of light in a vacuum.

Setting the propagation velocity will cause the dielectric constant to be set as appropriate.

**Group** Horizontal

**Syntax**     `HORizontal:DISTance:PVELocity <NR3>`  
`HORizontal:DISTance:PVELocity?`

**Related Commands**     [HORizontal:UNIts](#), [HORizontal:DISTance:DIElectric](#)

**Arguments**     NR3 is the propagation velocity of the material being tested, in meters per second.

**Examples**     `HORizontal:DISTance:PVELocity 202.12000E6` sets the propagation velocity for distance units conversion to  $2.0212 \times 10^8$  m/s.

`HORizontal:DISTance:PVELocity?` might return  
`HORizontal:DISTance_PVELocity 300.00000E6`, indicating that the propagation velocity value is set to  $3 \times 10^8$  m/s.

## **HORizontal:EXT10MHZref:FREQ**

This command sets or queries the external 10 MHz reference frequency. The command allows you to adjust the external 10 MHz reference frequency to ensure that the timebase locks when the Lock to Ext. 10 MHz mode is set using the `TDR:EXT10MHz` command.

**Group**     Horizontal

**Syntax**     `HORizontal:EXT10MHZref:FREQ <NR3>`  
`HORizontal:EXT10MHZref:FREQ?`

**Arguments**     NR3 is the external 10 MHz frequency setting. The range for this argument is  $1.0E7 \pm 0.005\%$  (9.9995 MHz through 10.0005 MHz). The resolution is 0.1 Hz. The default is 10 MHz.

**Examples**     `HORIZONTAL:EXT10MHZREF:FREQ 9.9995E6` sets the horizontal external 10 MHz frequency to 9.9995 MHz.

`HORIZONTAL:EXT10MHZREF:FREQ?` might return  
`HORIZONTAL:EXT10MHZREF:FREQ 10.000000000E+006`, indicating that the horizontal external 10 MHz frequency is set to the default value.

## HORizontal:FRAMescan:RESET (No Query Form)

This command (no query form) resets and restarts the FrameScan acquisition. Sending this command is equivalent to clicking the Reset button in the FrameScan section of the Pattern Sync/FrameScan dialog box.

<b>Group</b>	Horizontal
<b>Syntax</b>	HORizontal:FRAMescan:RESET
<b>Examples</b>	HORIZONTAL:FRAMESCAN:RESET resets and restarts the FrameScan acquisition.

## HORizontal:FRAMescan:SCANBits

This command sets or queries the number of bits or subframes to scan in the FrameScan mode. Sending this command is equivalent to setting a value in the Scan Bits control in the FrameScan section of the Pattern Sync/FrameScan Setup dialog box.

Scanning will begin at either the current horizontal bit location. The minimum value of bits to scan is 1 and the maximum value is  $2^{23}$  (8,288,608).

<b>Group</b>	Horizontal
<b>Syntax</b>	HORizontal:FRAMescan:SCANBits <NR1> HORizontal:FRAMescan:SCANBits?
<b>Arguments</b>	NR1 sets the number of bits to be scanned in FrameScan mode. The range is 1 through 8,288,608 ( $2^{23}$ ) bits.
<b>Examples</b>	HORIZONTAL:FRAMESCAN:SCANBITS 4000 sets the number of bits scanned in the FrameScan mode to 4000.  HORIZONTAL:FRAMESCAN:SCANBITS? might return HORIZONTAL:FRAMESCAN:SCANBITS 700, indicating the number of scan bits is set to 700.

## HORizontal:FRAMescan:STATE

This command sets or queries the FrameScan mode. Assuming that acquisitions are running, scanning starts as soon as you enable FrameScan. Sending this

command is equivalent to checking Enabled in the FrameScan section of the Pattern Sync/FrameScan Setup dialog box.

FrameScan acquisitions allow detailed display and analysis of individual, complete waveforms or of the bit sequences leading up to a failure. This ability to identify the specific patterns that cause the failures makes using FrameScan mode superior to traditional methods of analyzing pattern-dependent failures in high bit rate communications signals.

Refer to the online help for this instrument for additional information about FrameScan.

**Group** Horizontal

**Syntax** HORIZONTAL:FRAMESCAN:STATE { ON | OFF | NR1 }  
HORIZONTAL:FRAMESCAN:STATE?

- Arguments**
- ON enables the FrameScan mode.
  - OFF disables the FrameScan mode.
  - NR1 set to 0 disables the FrameScan mode; any other value enables the FrameScan mode.

**Examples** HORIZONTAL:FRAMESCAN:STATE ON enables the FameScan mode.  
HORIZONTAL:FRAMESCAN:STATE? might return  
HORIZONTAL:FRAMESCAN:STATE 0, indicating that the FrameScan mode is currently disabled.

---

**NOTE.** A query always returns a 0 or 1.

---

## HORIZONTAL:MAGNIFY<x>? (Query Only)

This query only command returns the timebase settings for the specified Mag timebase. The Mag timebase is specified by x, which can be 1 or 2.

**Group** Horizontal

**Syntax** HORIZONTAL:MAGNIFY<x>?

**Examples** HORIZONTAL:MAG1? might return HORIZONTAL:MAGNIFY1:SCALE  
1.05200000000E-009; POSITION 64.0380000000E-009;



RESOLUTION 5.2600000000E-012; RECORDLENGTH 2000;REFPOINT 0.0000000000; VIEW 1; TOLPOINT 74.5580000000E-009; TOFPOINT 64.0380000000E-009.

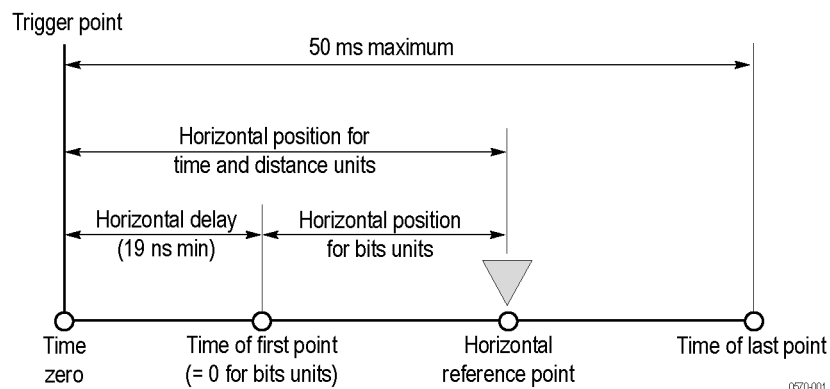
## HORizontal:MAGnify<x>:POSition

This command sets or queries the horizontal position for the specified Mag timebase. The Mag timebase is specified by x, which can be 1 or 2. Sending this command is equivalent to selecting a Mag timebase and entering a value in the Position control in the Timebase section of the Horizontal Setup dialog box or adjusting the front-panel Horizontal position control.

This command sets a delay time (or distance) for the specified Mag channel waveforms between the trigger event and the horizontal reference point for the Mag timebase, which is set with the HORizontal:MAG<x>:REFPoint command. The minimum position you can set is limited to one of the following settings, depending on the current hardware configuration:

- If there are no sampling modules on extender cables, the minimum horizontal position is set to the insertion delay of the module having the highest insertion delay specification. This is the typical value, not the actual hardware minimum value (insertion delay, or the minimum time from an external trigger event to the first acquired sample).
- If there are one or more sampling modules on extender cables, then the minimum horizontal position is set to the combined insertion delay and extender delay of the module having the highest combination of insertion delay specification plus attached extender delay. The minimum horizontal position governs the smallest horizontal position you can set. Changing the main timebase position can affect the position of each Mag timebase; however, Mag timebase positions cannot affect the main timebase position.
- Mag acquisitions are contained within the Main timebase acquisitions.

Refer to the illustration below for a diagram of the relationships of horizontal position, horizontal reference point, time of first point, and time of last point.



The position of the Mag timebase is constrained to be in the domain of the Main timebase. Changing the Main timebase position can affect the position of each Mag timebase, but a Mag timebase position cannot affect the Main timebase position.

<b>Group</b>	Horizontal
<b>Syntax</b>	<code>HORizontal:MAGnify&lt;x&gt;:POSition &lt;NR3&gt;</code> <code>HORizontal:MAGnify&lt;x&gt;:POSition?</code>
<b>Related Commands</b>	<a href="#">HORizontal:MAGnify&lt;x&gt;:REFPoint</a> , <a href="#">HORizontal:MAGnify&lt;x&gt;:TOFPoint?</a> , <a href="#">HORizontal:MAGnify&lt;x&gt;:TOLPoint?</a>
<b>Arguments</b>	NR3 is the horizontal position value for the specified Mag timebase. The range is from a typical value of 19 ns to 50 ms. The minimum value will be equal to or greater than the insertion delay (see bulleted item above).
<b>Examples</b>	<code>HORIZONTAL:MAG1:POSITION 5.0E-8</code> attempts to set the horizontal position for the Mag1 timebase to 50 ns. (If the Main timebase does not cover the time window, the Mag horizontal position will be coerced.)  <code>HORIZONTAL:MAG2:POSITION?</code> might return <code>HORIZONTAL:MAGNIFY2:POSITION 84.0000000000E-009</code> , indicating that the horizontal position for the MAG2 timebase is set to a time of 84 ns relative to the trigger.

## **HORizontal:MAGnify<x>:RECORDlength**

This command sets or queries the record length for the specified Mag timebase. The Mag timebase is specified by x, which can be 1 or 2. Sending this command is equivalent to selecting a Mag timebase and then setting a value for the Record Length control in the Timebase section of the Horizontal Setup dialog box.

The instrument displays the waveform within the 10 horizontal graticule divisions. The default is 4000 samples.

<b>Group</b>	Horizontal
<b>Syntax</b>	<code>HORizontal:MAGnify&lt;x&gt;:RECORDlength &lt;NR1&gt;</code> <code>HORizontal:MAGnify&lt;x&gt;:RECORDlength?</code>
<b>Related Commands</b>	<a href="#">HORizontal:MAGnify&lt;x&gt;:SCALE</a> , <a href="#">HORizontal:MAGnify&lt;x&gt;:RESolution?</a>

**Arguments** NR1 is the record length. Valid record lengths are 50, 100, 250, 500, 1000, 2000, and 4000 samples.

**Examples** `HORIZONTAL:MAG2:RECORDLENGTH 1000` sets the Mag 2 timebase record length to 1000 samples.

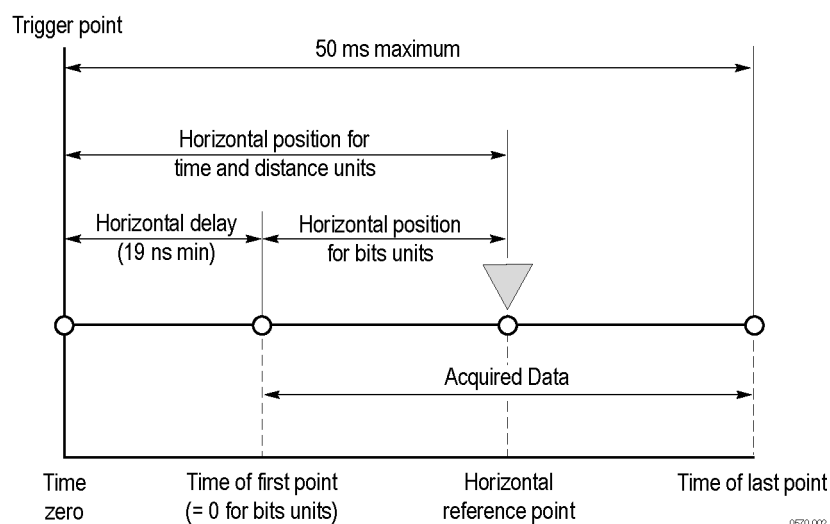
`HORIZONTAL:MAG<x>:RECORDLENGTH?` might return  
`HORIZONTAL:MAGNIFY1:RECORDLENGTH 4000`, indicating that the record length for the Mag 1 timebase is set to 4000 samples.

## HORizontal:MAGnify<x>:REFPoint

This command sets or queries the horizontal reference point for the specified Mag timebase. The Mag timebase is specified by x, which can be 1 or 2. The horizontal reference point is the horizontal position around which the `HORizontal:MAG<x>:SCALE` command contracts or expands the waveform, which affects the acquired and observable horizontal window of data. Also, it represents the point that determines the horizontal position of the record from the trigger. The horizontal reference position is set as a fraction of the record; 0.5 (50% of the record) is the default.

This command is equivalent to selecting a Mag timebase and setting a value in the Horizontal Reference control in the Timebase section of the Horizontal Setup dialog box or moving the Horizontal Reference Point indicator at the top of the Mag view.

Refer to the illustration below for a diagram of the relationships of horizontal reference point, time of first point, time of last point, and horizontal position.



**Group** Horizontal

**Syntax**      `HORizontal:MAGnify<x>:REFPoint <NR2>`  
`HORizontal:MAGnify<x>:REFPoint?`

**Related Commands**    [HORizontal:MAGnify<x>:SCAle](#)

**Arguments**      NR2 is the fraction of the record at which the horizontal reference point is set. The range is 0 through 1 (corresponding to 0% through 100% of record); the default is 0.5. The resolution is 0.01.

**Examples**      `HORIZONTAL:MAG1:REFPOINT 2.5E-1` sets the horizontal reference point for the Mag1 timebase to 0.25 (25%). This is the point around which the `HORizontal:MAG:SCAle` command contracts or expands the waveform.

`HORIZONTAL:MAG2:REFPOINT?` might return  
`HORIZONTAL:MAGNIFY2:REFPOINT .500000000000`, indicating that the horizontal reference point for the Mag2 timebase is set to 0.5 (50%), which is the default.

## **HORizontal:MAGnify<x>:RESolution? (Query Only)**

This query only command returns the current resolution per sample of the specified Mag timebase. The Mag timebase is specified by x, which can be 1 or 2. The resolution for the selected timebase is displayed in the Resolution field in the Timebase section of the Horizontal Setup dialog box. You cannot directly set the resolution. Instead, use the `HORizontal:MAG<x>:SCAle` command to set the horizontal scale and the `HORizontal:MAG<x>:RECORDLength` command to set the record length. Resolution is always displayed in time.

**Group**      Horizontal

**Syntax**      `HORizontal:MAGnify<x>:RESolution?`

**Related Commands**    [HORizontal:MAGnify<x>:SCAle](#), [HORizontal:MAGnify<x>:RECORDlength](#)

**Examples**      `HORIZONTAL:MAG1:RESOLUTION?` might return  
`HORIZONTAL:MAGNIFY1:RESOLUTION 6.31000000000E-012`, indicating that the resolution is 6.31 ps.

## HORizontal:MAGnify<x>:SCALE

This command sets or queries the scale (time per division) for the specified Mag timebase. The Mag timebase is specified by x, which can be 1 or 2. Sending this command is equivalent to selecting a Mag timebase and adjusting the Scale control in the Timebase section of the Horizontal Setup dialog box (or adjusting the Horizontal SCALE front-panel knob).

Channel waveforms display in the Mag timebases with the horizontal scale set for the timebase. Each channel timebase, Main, Mag1, and Mag2, uses its own horizontal scale. All channel waveforms in an individual timebase share the timebase scale. The horizontal scale, together with the record length setting, determines the sample interval. Main, Mag1, and Mag2 timebase scales, positions, and record lengths (resolution) are independent of one another; however, acquisitions on Mag timebases must be totally contained within the Main timebase.

**Group** Horizontal

**Syntax** `HORizontal:MAGnify<x>:SCALE <NR3>`  
`HORizontal:MAGnify<x>:SCALE?`

**Related Commands** [HORizontal:MAGnify<x>:RECOrdlength](#), [HORizontal:MAGnify<x>:RESolution?](#), [HORizontal:MAGnify<x>:REFPoint](#), [HORizontal:MAGnify<x>:POSition](#)

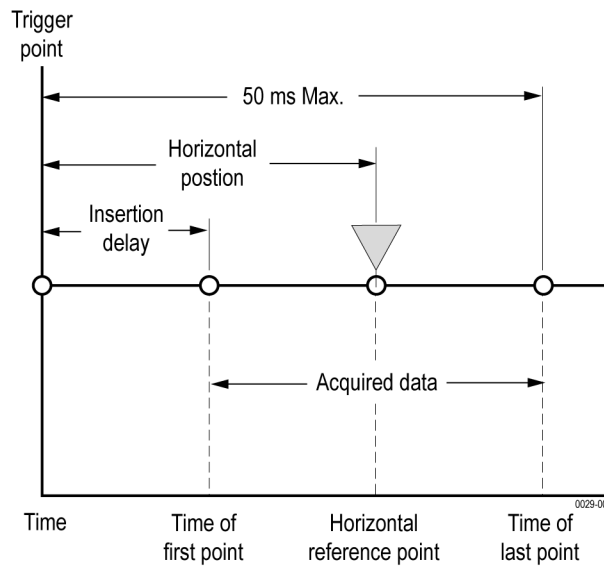
**Arguments** NR3 is the horizontal time per division for the specified Mag timebase. The range is 1 ps per division to 5 ms per division.

**Examples** `HORIZONTAL:MAG1:SCALE 2E-6` sets the MAG1 scale to 2  $\mu$ s per division.  
`HORIZONTAL:MAG2:SCALE?` might return `HORIZONTAL:MAGNIFY2:SCALE 2.0000000000E-009`, indicating that the Mag2 timebase is currently set to 2 ns per division.

## HORizontal:MAGnify<x>:TOFPoint? (Query Only)

This query only command returns time of first point for the specified Mag timebase, which is the time interval between the trigger point and the first point in the specified Mag timebase record (waveform). The Mag timebase is specified by x, which can be 1 or 2.

Refer to the illustration below for a diagram of the relationships of time of first point, time of last point, horizontal position, and horizontal reference point.



**Group** Horizontal

**Syntax** `HORizontal:MAGnify<x>:TOFPoint?`

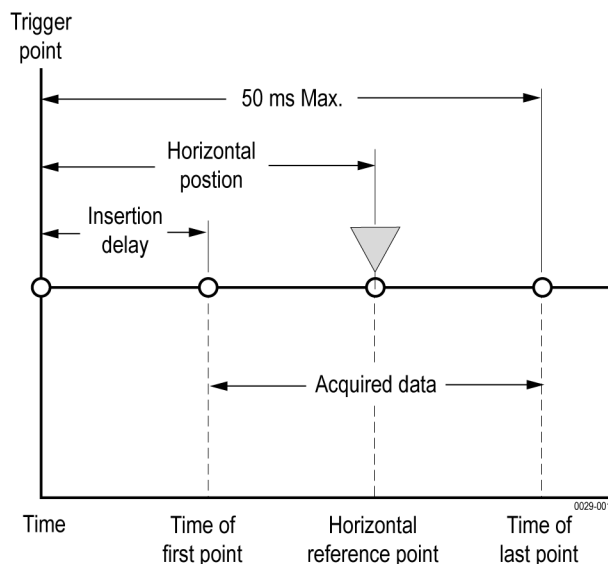
**Related Commands** [HORizontal:MAGnify<x>:POSition](#), [HORizontal:MAGnify<x>:REFPoint](#), [HORizontal:MAGnify<x>:SCALE](#)

**Examples** `HORIZONTAL:MAG1:TOFPOINT?` might return `HORIZONTAL:MAG1:TOFPOINT 50.000000000E-9`, indicating that the time interval between the trigger point and the first point in the Mag1 timebase record is 50 ns.

## **HORizontal:MAGnify<x>:TOLPoint? (Query Only)**

This query only command returns time of last point for the specified Mag timebase, which is the time interval between the trigger point and the last point in the specified Mag timebase record (waveform).

Refer to the illustration below for a diagram of the relationships of time of last point, time of first point, horizontal position, and horizontal reference point.



**Group** Horizontal

**Syntax** `HORizontal:MAGnify<x>:TOLPoint?`

**Examples** `HORIZONTAL:MAG2:TOLPOINT?` might return `HORIZONTAL:MAG2:TOLPOINT 100.0000000000E-6`, indicating that the time interval between the trigger point and the last point in the Mag2 timebase record is 100  $\mu$ s.

## **HORizontal:MAGnify<x>:VIEW**

This command sets or queries whether the specified Mag timebase is acquired and displayed. The Mag timebase is specified by *x*, which can be 1 or 2. Sending this command is equivalent to clicking the Mag1 or Mag2 icon in the timebase selector drop down box located below the graticule area of the display, or pressing the equivalent VIEW buttons on the front panel.

**Group** Horizontal

**Syntax** `HORizontal:MAGnify<x>:VIEW { ON | OFF | NR1 }`  
`HORizontal:MAGnify<x>:VIEW?`

- Arguments**
- ON turns on the acquisition and display of the specified Mag timebase.
  - OFF turns off the acquisition and display of the specified Mag timebase.
  - NR1 set to 0 disables the specified Mag timebase view; any other integer enables the specified Mag timebase view.

**Examples** HORIZONTAL:MAG2:VIEW ON displays the Mag2 timebase.  
 HORIZONTAL:MAG1:VIEW? might return HORIZONTAL:MAGNIFY1:VIEW 0, indicating that the display of the Mag1 timebase is disabled.

---

**NOTE.** A query always returns a 0 or 1.

---

## HORizontal:MAIn? (Query Only)

This query only command returns the settings of the main time base.

**Group** Horizontal

**Syntax** HORizontal:MAIn?

**Examples** HORIZONTAL:MAIN? might return HORIZONTAL:MAIN:SCALE 2.0000000000E-009; POSITION 20.3000000000E-009; RESOLUTION 5.0000000000E-012; RECORDLENGTH 4000; REFPOINT 0.0000000000; TOLPOINT 40.3000000000E-009; TOFPOINT 20.3000000000E-009.

## HORizontal:MAIn:BITS:POSition

This command sets or queries the horizontal position for the main timebase in bits. Sending this command is equivalent to selecting MainTB and entering a value in the Position control in the Timebase section of the Horizontal Setup dialog box or adjusting the front-panel Horizontal position control when the horizontal units are set to BITS. The equivalent position in time can be computed as  $\text{time} = (\text{bits} / \text{bit rate}) + \text{insertion delay}$ . The bit rate is the rate specified by the HORIZONTAL:BITS:BITRATE command.

**Group** Horizontal

**Syntax** HORizontal:MAIn:BITS:POSition NR3  
 HORizontal:MAIn:BITS:POSition?



**Arguments** ■ NR3 is the horizontal position value in bits for the Main timebase. The range is from 0 to 50 ms \* bit rate.

**Examples** `HORIZONTAL:MAIN:BITS:POSITION 10.0` sets the horizontal position for the Main timebase to 10 bits (equivalent to 20 ns if the bit rate is 10e9 and the insertion delay is 19 ns).

`HORIZONTAL:MAIN:BITS:POSITION?` might return 20.0, indicating that the horizontal position for the Main timebase is set to 20 bits (equivalent to 21 ns if the bit rate is 10e9 and the insertion delay is 19 ns).

## HORizontal:MAIn:BITS:SCALE

This command sets or queries the scale (bits per division) for the Main timebase. Sending this command is equivalent to selecting MainTB and adjusting the Scale control in the Timebase section of the Horizontal Setup dialog box when the horizontal units are set to BITS.

**Group** Horizontal

**Syntax** `HORizontal:MAIn:BITS:SCALE NR3`  
`HORizontal:MAIn:BITS:SCALE?`

**Arguments** ■ NR3 is the horizontal bits per screen for the Main timebase. The range is 1 ps \* 10 divs \* bit rate to 5 ms \* 10 divs \* bit rate.

**Examples** `HORIZONTAL:MAIN:BITS:SCALE 10` sets the time span for the screen to 10 bits when the horizontal units are set to bits and the display scale is set to per screen.

`HORIZONTAL:MAIN:BITS:SCALE?` might return 20 indicating that the time span for the screen is 20 bits when the horizontal units are set to bits and the display scale is set to per screen.

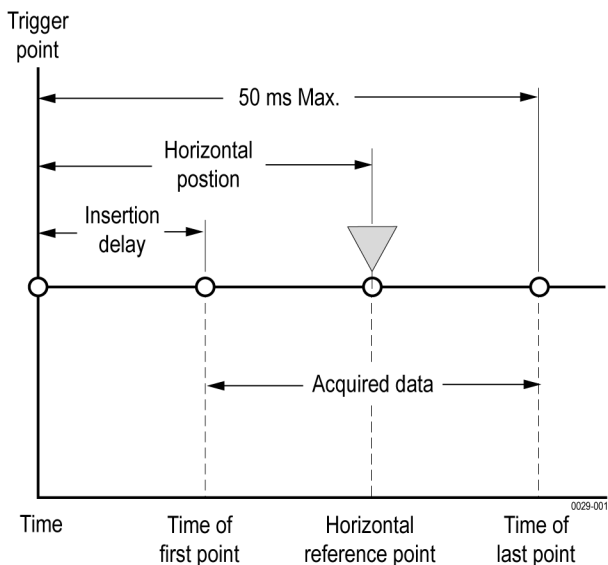
## HORizontal:MAIn:POSition

This command sets or queries the horizontal position for the Main timebase. Sending this command is equivalent to selecting MainTB and entering a value in the Position control in the Timebase section of the Horizontal Setup dialog box or adjusting the front-panel Horizontal position control.

This command sets a global delay time (or distance) for all channel waveforms between the trigger event and the horizontal reference point for the Main timebase, which is set with the `HORizontal:MAIn:REFPoint` command. The minimum position you can set is limited to one of two settings, depending on the current hardware configuration:

- If there are no sampling modules on extender cables, the minimum horizontal position is set to the insertion delay of the module having the highest insertion delay specification. This is the typical value, not the actual hardware minimum value (insertion delay, or the minimum time from an external trigger event to the first acquired sample).
- If there are one or more sampling modules on extender cables, then the minimum horizontal position is set to the combined insertion delay and extender delay of the module having the highest combination of insertion delay specification plus attached extender delay. The minimum horizontal position governs the smallest horizontal position you can set. Changing the main timebase position can affect the position of each Mag timebase; however, Mag timebase positions cannot affect the main timebase position.

Refer to the illustration below for a diagram of the relationships of horizontal position, horizontal reference point, time of first point, and time of last point.



**Group** Horizontal

**Syntax** `HORizontal:MAIn:POSition <NR3>`  
`HORizontal:MAIn:POSition?`

<b>Related Commands</b>	<a href="#">HORizontal:MAIn:REFPoint</a> , <a href="#">HORizontal:MAIn:TOFPoint?</a> , <a href="#">HORizontal:MAIn:TOLPoint?</a>
<b>Arguments</b>	NR3 is the horizontal position value for the Main timebase. The range is from a typical value of 19 ns to 50 ms. The minimum value will be equal to or greater than the insertion delay (see bulleted item above).
<b>Examples</b>	<p><code>HORIZONTAL:MAIN:POSITION 1.0E-7</code> sets the horizontal position for the Main timebase to 100 ns.</p> <p><code>HORIZONTAL:MAIN:POSITION?</code> might return <code>HORIZONTAL:MAIN:POSITION 95.0000000000E-009</code>, indicating that the horizontal position for the Main timebase is set to 95 ns.</p>

## HORizontal:MAIn:RECORDlength

This command sets or queries the record length for the Main timebase. This is equivalent to selecting MainTB and then setting a value for the Record Length control in the Timebase section of the Horizontal Setup dialog box.

The instrument displays the waveform within the 10 horizontal divisions on screen. The default is 4000 samples.

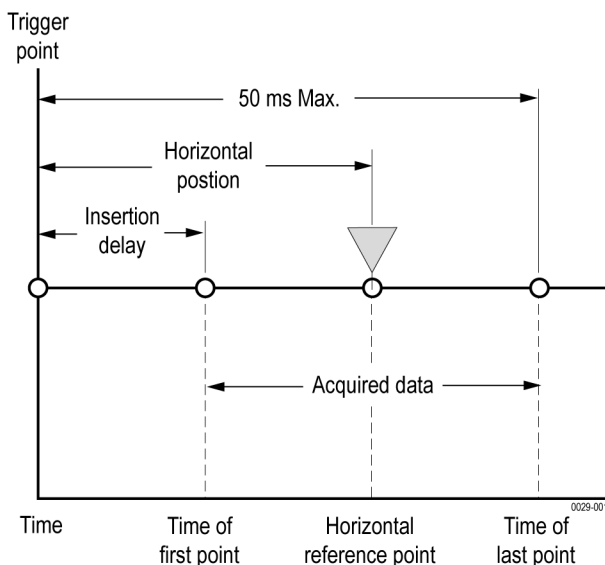
<b>Group</b>	Horizontal
<b>Syntax</b>	<code>HORizontal:MAIn:RECORDlength &lt;NR1&gt;</code> <code>HORizontal:MAIn:RECORDlength?</code>
<b>Related Commands</b>	<a href="#">HORizontal:MAIn:SCALE</a> , <a href="#">HORizontal:MAIn:RESolution?</a>
<b>Arguments</b>	NR1 is the record length. Valid record lengths are 50, 100, 250, 500, 1000, 2000, 4000, 8000, and 16000 samples.
<b>Examples</b>	<p><code>HORIZONTAL:MAIN:RECORDLENGTH 500</code> sets the record length for the Main timebase to 500 samples.</p> <p><code>HORIZONTAL:MAIN:RECORDLENGTH ?</code> might return <code>HORIZONTAL:MAIN:RECORDLENGTH 1000</code>, indicating that the record length for the Main timebase is set to 1000 samples.</p>

## HORizontal:MAIn:REFPoint

This command sets or queries the horizontal reference point for the Main timebase. The horizontal reference point is the horizontal position around which the HORizontal:MAIn:SCAle command contracts or expands the waveform, which affects the acquired and observable horizontal window of data. The horizontal reference position is set as a fraction of the record; 0.5 (50% of the record) is the default.

This command is equivalent to setting a value in the Horizontal Reference control in the Timebase section of the Horizontal Setup dialog box or moving the Horizontal Reference Point indicator at the top of the Main view.

Refer to the illustration below for a diagram of the relationships of horizontal reference point, time of first point, time of last point, and horizontal position.



**Group** Horizontal

**Syntax** `HORizontal:MAIn:REFPoint <NR2>`  
`HORizontal:MAIn:REFPoint?`

**Related Commands** [HORizontal:MAIn:POStion](#)

**Arguments** NR2 is the fraction of the record at which the horizontal reference point is set. The range is 0 through 1 (corresponding to 0% through 100% of record); the default is 0.5. The resolution is 0.01.

**Examples** `HORIZONTAL:MAIN:REFPOINT 2.5E-1` sets the horizontal reference point for the Main timebase to 0.25 (25%). This is the point around which the `HORizontal:MAIn:SCAle` command contracts or expands the waveform.

`HORIZONTAL:MAIN:REFPOINT?` might return `HORIZONTAL:MAIN:REFPOINT .50000000000`, indicating that the horizontal reference point for the Main timebase is set to 0.5 (50%), which is the default.

## HORizontal:MAIn:RESolution? (Query Only)

This query only command returns the current resolution per sample of the Main timebase. This value is the same as the value displayed in the Resolution field in the Timebase section of the Horizontal Setup dialog box. You cannot directly set the resolution. Instead, use the `HORizontal:MAIn:SCAle` command to set the horizontal scale and the `HORizontal:MAIn:RECORDLength` command to set the record length. Resolution is always displayed in time units.

**Group** Horizontal

**Syntax** `HORizontal:MAIn:RESolution?`

**Related Commands** [HORizontal:MAIn:SCAle](#), [HORizontal:MAIn:RECORDlength](#)

**Examples** `HORIZONTAL:MAIN:RESOLUTION?` might return `HORIZONTAL:MAIN:RESOLUTION 25.0000000000E-012`, indicating that the horizontal resolution for the Main timebase is 25 ps.

## HORizontal:MAIn:SCAle

This command sets or queries the scale (time per division) for the Main timebase. Sending this command is equivalent to selecting MainTB and adjusting the Scale control in the Timebase section of the Horizontal Setup dialog box.

The horizontal scale that you set with this command is common to all channel waveforms using the Main timebase. The horizontal scale, together with the record length setting, determines the sample interval.

**Group** Horizontal

**Syntax** `HORizontal:MAIn:SCAle <NR3>`  
`HORizontal:MAIn:SCAle?`

**Related Commands**    [HORizontal:MAIn:RECOrdlength](#), [HORizontal:MAIn:RESolution?](#)

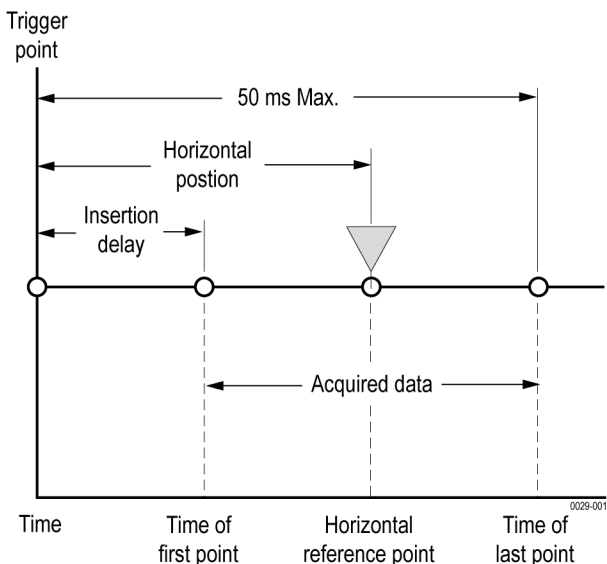
**Arguments**    NR3 is the horizontal time per division for the Main timebase. The range is 1 ps per division to 5 ms per division.

**Examples**    `HORIZONTAL:MAIN:SCALE 2.0E-9` sets the main scale to 2 ns per division.  
`HORIZONTAL:MAIN:SCALE?` might return `HORIZONTAL:MAIN:SCALE 10.000000000E-009`, indicating that the Main timebase is currently set to 10 ns per division.

### HORizontal:MAIn:TOFPoint? (Query Only)

This query only command returns time of first point, which is the time interval between the trigger point and the first point in a Main timebase record (waveform).

Refer to the illustration below for a diagram of the relationships of time of first point, time of last point, horizontal position, and horizontal reference point.



**Group**    Horizontal

**Syntax**    `HORizontal:MAIn:TOFPoint?`

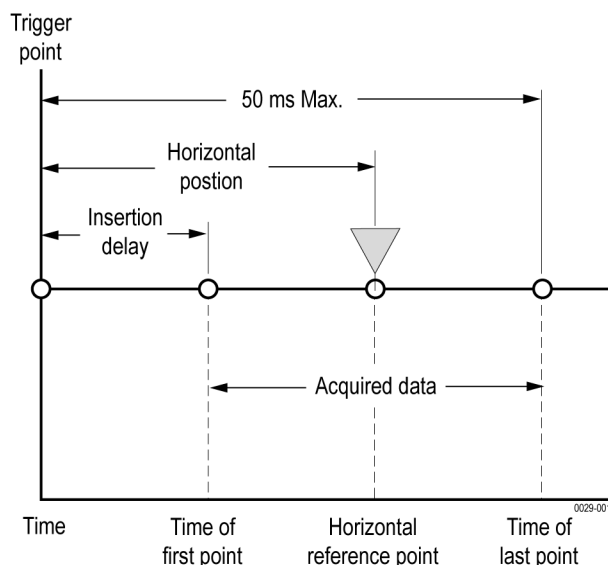
**Related Commands**    [HORizontal:MAIn:POStion](#), [HORizontal:MAIn:REFPoint](#), [HORizontal:MAIn:SCALE](#)

**Examples** `HORIZONTAL:MAIN:TofPOINT?` might return `HORIZONTAL:MAIN:TofPOINT 20.3000000000E-009`, indicating that the time interval between the trigger point and the first point in the Main timebase record is 20.3 ns.

## HORizontal:MAIn:TOLPoint? (Query Only)

This query only command returns time of last point, which is the time interval between the trigger point and the last point in a Main timebase record (waveform).

Refer to the illustration below for a diagram of the relationships of time of last point, time of first point, horizontal position, and horizontal reference point.



**Group** Horizontal

**Syntax** `HORizontal:MAIn:TOLPoint?`

**Examples** `HORIZONTAL:MAIN:TOLPOINT?` might return `HORIZONTAL:MAIN:TOLPOINT 40.3000000000E-009`, indicating that the time interval between the trigger point and the last point in the Main timebase record is 40.3 ns.

## HORizontal:MATH<x>:MAGnify<x>:POSition? (Query Only)

This query only command returns the horizontal position of the specified math waveform using the specified Mag timebase. The math waveform is specified by *x*, which can be 1 through 8; the Mag timebase is specified by *x*, which can be 1 or 2.

**Group** Horizontal

**Syntax** `HORizontal:MATH<x>:MAGnify<x>:POSition?`

**Examples** `HORIZONTAL:MATH1:MAG2:POSITION?` might return `HORIZONTAL:MATH1:MAGNIFY2:POSITION 19.0000000000E-009`, indicating that the horizontal position of the Math 2 waveform using the MAG2 timebase is set to the minimum time of 19 ns.

## **HORizontal:MATH<x>:MAGnify<x>:RECOrdlength? (Query Only)**

This query only command returns the record length of the specified Mag timebase for the specified math waveform. The value of the record length can be 50, 100, 250, 500, 1000, 2000, and 4000 samples. It results from normalizing the sources of a math waveform.

**Group** Horizontal

**Syntax** `HORizontal:MATH<x>:MAGnify<x>:RECOrdlength?`

**Related Commands** [HORizontal:MATH<x>:MAGnify<x>:SCAlE?](#), [HORizontal:MATH<x>:MAGnify<x>:RESolution?](#)

**Examples** `HORIZONTAL:MATH5:MAG2:RECORDLENGTH?` might return `HORIZONTAL:MATH5:MAGNIFY2:RECORDLENGTH 1000`, indicating that the record length of the Mag 2 timebase is set to 1000 samples for the Math 5 waveform.

## **HORizontal:MATH<x>:MAGnify<x>:REFPpoint? (Query Only)**

This query only command returns the horizontal reference point of the specified math waveform using the specified magnification timebase. The math waveform is specified by x, which can be 1 through 8. The magnification timebase can be 1 or 2.

The horizontal reference point is the horizontal position around which the `HORizontal:MAGnify<x>:SCAlE` command contracts or expands the waveform. This affects the acquired and observable horizontal window of data. The horizontal reference position is set as a fraction of the record (0.5 = 50%).



<b>Group</b>	Horizontal
<b>Syntax</b>	<code>HORizontal:MATH&lt;x&gt;:MAGnify&lt;x&gt;:REFPoint?</code>
<b>Related Commands</b>	<a href="#">HORizontal:MATH&lt;x&gt;:MAGnify&lt;x&gt;:SCALE?</a> , <a href="#">HORizontal:MATH&lt;x&gt;:MAGnify&lt;x&gt;:RESolution?</a>
<b>Examples</b>	<code>HORizontal:MATH5:MAGnify2:REFPoint?</code> might return <code>HORIZONTAL:MATH5:MAGNIFY2:REFPOINT 5.0000000000E-001</code> , indicating that this Math waveform will be scaled around the 50% point of the Mag2 window.

## **HORizontal:MATH<x>:MAGnify<x>:RESolution? (Query Only)**

This query only command returns the resolution per sample of the specified Mag timebase for the specified math waveform. The math waveform is specified by x, which can be 1 through 8; the Mag timebase is specified by x, which can be 1 or 2. Resolution is always displayed in time units.

<b>Group</b>	Horizontal
<b>Syntax</b>	<code>HORizontal:MATH&lt;x&gt;:MAGnify&lt;x&gt;:RESolution?</code>
<b>Related Commands</b>	<a href="#">HORizontal:MATH&lt;x&gt;:MAGnify&lt;x&gt;:SCALE?</a> , <a href="#">HORizontal:MATH&lt;x&gt;:MAGnify&lt;x&gt;:RECOrdlength?</a>
<b>Examples</b>	<code>HORIZONTAL:MATH1:MAG1:RESOLUTION?</code> might return <code>HORIZONTAL:MATH1:MAGNIFY1:RESOLUTION 1.0000000000E-009</code> , indicating that the resolution of the Mag1 timebase for the Math1 waveform is 1 ns.

## **HORizontal:MATH<x>:MAGnify<x>:SCALE? (Query Only)**

This query only command returns the scale (time per division) of the specified Mag timebase for the specified math waveform. The math waveform is specified by x, which can be 1 through 8; the Mag timebase is specified by x, which can be 1 or 2.

Math waveforms derive their timebase settings from the waveform sources that define them. Math scale is inherited from the live channel source of the

math waveform or the saved reference waveforms, if there are no live channel waveforms in the math expression.

**Group** Horizontal

**Syntax** `HORizontal:MATH<x>:MAGnify<x>:SCALE?`

**Related Commands** [HORizontal:MAGnify<x>:RECORDlength](#), [HORizontal:MAGnify<x>:RESolution?](#)

**Examples** `HORIZONTAL:MATH2:MAG2:SCALE?` might return `HORIZONTAL:MATH1:MAGNIFY1:SCALE 30.000000000000E-009`, indicating that the Mag1 timebase is currently set to 30 ns per division for the Math 1 waveform.

## **HORizontal:MATH<x>:MAGnify<x>:TOFPoint? (Query Only)**

This query only command returns the time of first point of the specified math waveform using the specified Mag timebase. The math waveform is specified by x, which can be 1 through 8; the Mag timebase is specified by x, which can be 1 or 2. The time of first point is the time interval between the trigger point and the first point in the specified Mag timebase record (waveform).

**Group** Horizontal

**Syntax** `HORizontal:MATH<x>:MAGnify<x>:TOFPoint?`

**Related Commands** [HORizontal:MATH<x>:MAGnify<x>:POSITION?](#), [HORizontal:MATH<x>:MAGnify<x>:SCALE?](#)

**Examples** `HORIZONTAL:MATH4:MAG1:TOFPOINT?` might return `HORIZONTAL:MATH4:MAGNIFY1:TOFPOINT 50.000000000000E-009`, indicating that the time interval between the trigger point and the first point in the Math 4 waveform record using the Mag1 timebase is 50 ns.

## **HORizontal:MATH<x>:MAGnify<x>:TOLPoint? (Query Only)**

This query only command returns time of last point of the specified math waveform using the specified Mag timebase. The math waveform is specified by

x, which can be 1 through 8; the Mag timebase is specified by x, which can be 1 or 2. The time of last point is the time interval between the trigger point and the last point in the specified Mag timebase record (waveform).

**Group** Horizontal

**Syntax** `HORizontal:MATH<x>:MAGnify<x>:TOLPoint?`

**Related Commands** [HORizontal:MATH<x>:MAGnify<x>:POSition?](#), [HORizontal:MATH<x>:MAGnify<x>:SCALE?](#)

**Examples** `HORIZONTAL:MATH2:MAG2:TOLPOINT?` might return `HORIZONTAL:MAGNIFY2:TOLPOINT 1.0000000000E-9`, indicating that the time interval between the trigger point and the last point in the Math 2 waveform record using the Mag2 timebase is 1 ns.

## **HORizontal:MATH<x>:MAIn:POSition? (Query Only)**

This query only command returns the horizontal position of the specified math waveform using the Main timebase. The math waveform is specified by x, which can be 1 through 8.

**Group** Horizontal

**Syntax** `HORizontal:MATH<x>:MAIn:POSition?`

**Related Commands** [HORizontal:MATH<x>:MAIn:REFPoint?](#), [HORizontal:MATH<x>:MAIn:TOFPoint?](#), [HORizontal:MATH<x>:MAIn:TOLPoint?](#)

**Examples** `HORIZONTAL:MATH8:MAIN:POSITION?` might return `HORIZONTAL:MATH8:MAIN:POSITION 19.0000000000E-009`, indicating that the horizontal position of the Math 8 waveform using the Main timebase is set to the minimum time of 19 ns.

## **HORizontal:MATH<x>:MAIn:RECOrdlength? (Query Only)**

This query only command returns the record length of the Main timebase for the specified math waveform. The math waveform is specified by x, which can be 1 through 8.

Record lengths can be 50, 100, 250, 500, 1000, 2000, 4000, 8000, and 16000 samples. It results from normalizing the sources of a math waveform.

**Group** Horizontal

**Syntax** `HORizontal:MATH<x>:MAIn:RECOrdlength?`

**Related Commands** [HORizontal:MATH<x>:MAIn:SCAle?](#), [HORizontal:MATH<x>:MAIn:RESolution?](#)

**Examples** `HORIZONTAL:MATH4:MAIN:RECORDLENGTH?` might return `HORIZONTAL:MATH4:MAIN:RECORDLENGTH 1000`, indicating that the record length for the Main timebase is set to 1000 samples for the Math 4 waveform.

## **HORizontal:MATH<x>:MAIn:REFPoint? (Query Only)**

This query only command returns the horizontal reference point of the specified math waveform using the Main timebase. The math waveform is specified by *x*, which can be 1 through 8. The horizontal reference point is the horizontal position around which the `HORizontal:MAIn:SCAle` command contracts or expands the waveform, which affects the acquired and observable horizontal window of data. The horizontal reference position is set as a fraction of the record; 0.5 (50% of the record) is the default.

**Group** Horizontal

**Syntax** `HORizontal:MATH<x>:MAIn:REFPoint?`

**Related Commands** [HORizontal:MATH<x>:MAIn:SCAle?](#)

**Examples** `HORIZONTAL:MATH3:MAIN:REFPOINT?` might return `HORIZONTAL:MATH3:MAIN:REFPOINT .5000000000`, indicating that the horizontal reference point for the Math 3 waveform using the Main timebase is set to 0.5 (50%).

## HORizontal:MATH<x>:MAIn:RESolution? (Query Only)

This query only command returns the current resolution per sample of the Main timebase for the specified math waveform. The math waveform is specified by x, which can be 1 through 8. Resolution is always displayed in time units.

**Group** Horizontal

**Syntax** HORizontal:MATH<x>:MAIn:RESolution?

**Related Commands** [HORizontal:MATH<x>:MAIn:SCAle?](#), [HORizontal:MATH<x>:MAIn:RECOrdlength?](#)

**Examples** HORIZONTAL:MATH2:MAIN:RESOLUTION? might return  
HORIZONTAL:MATH1:MAIN:RESOLUTION 1.0000000000E-009,  
indicating that the Main timebase resolution is at 1 ns for the Math2 waveform.

## HORizontal:MATH<x>:MAIn:SCAle? (Query Only)

This query only command returns the scale (time per division) of the Main timebase for the specified math waveform. The math waveform is specified by x, which can be 1 through 8.

**Group** Horizontal

**Syntax** HORizontal:MATH<x>:MAIn:SCAle?

**Related Commands** [HORizontal:MATH<x>:MAIn:RECOrdlength?](#), [HORizontal:MATH<x>:MAIn:RESolution?](#)

**Examples** HORIZONTAL:MATH4:MAIN:SCALE? might return  
HORIZONTAL:MATH1:MAIN:SCALE 1.0000000000E-009, indicating that the  
Main timebase for the Math4 waveform is set to 1 ns per division.

## HORizontal:MATH<x>:MAIn:TOFPoInt? (Query Only)

This query only command returns the time of first point of the Main timebase record, which is the time interval between the trigger point and the first point in

the record, for the specified math waveform. The math waveform is specified by *x*, which can be 1 through 8.

**Group** Horizontal

**Syntax** `HORizontal:MATH<x>:MAIn:TOFPoint?`

**Related Commands** [HORizontal:MATH<x>:MAIn:POSition?](#), [HORizontal:MATH<x>:MAIn:REFPoint?](#), [HORizontal:MATH<x>:MAIn:SCAle?](#)

**Examples** `HORIZONTAL:MATH3:MAIN:TOFPOINT?` might return `HORIZONTAL:MATH3:MAIN:TOFPOINT 100.00000000000E-9`, indicating that the time of first point for the Main timebase record for the Math3 waveform is 100 ns.

## **HORizontal:MATH<x>:MAIn:TOLPoint? (Query Only)**

This query only command returns the time of last point of the Main timebase record, which is the time interval between the trigger point and the last point in the record, for the specified math waveform. The math waveform is specified by *x*, which can be 1 through 8.

**Group** Horizontal

**Syntax** `HORizontal:MATH<x>:MAIn:TOLPoint?`

**Related Commands** [HORizontal:MATH<x>:MAIn:POSition?](#), [HORizontal:MATH<x>:MAIn:REFPoint?](#), [HORizontal:MATH<x>:MAIn:SCAle?](#)

**Examples** `HORIZONTAL:MATH3:MAIN:TOLPOINT?` might return `HORIZONTAL:MATH3:MAIN:TOLPOINT 36.00000000000E-009`, indicating that the time of last point for the Main timebase record for the Math3 waveform is 36 ns.

## **HORizontal:REF<x>:MAGnify<x>:POSition? (Query Only)**

This query only command returns the horizontal position of the specified Mag timebase for the specified Reference waveform. The Reference waveform is

specified by x, which can be 1 through 8; the Mag timebase is specified by x, which can be 1 or 2.

**Group** Horizontal

**Syntax** `HORizontal:REF<x>:MAGnify<x>:POSition?`

**Related Commands** [HORizontal:REF<x>:MAGnify<x>:TOFPoint?](#), [HORizontal:REF<x>:MAGnify<x>:TOLPoint?](#)

**Examples** `HORIZONTAL:REF1:MAG2:POSITION?` might return `HORIZONTAL:REF1:MAGNIFY2:POSITION 19.00000000000E-009`, indicating that the horizontal position for the Ref 1 waveform using the MAG2 timebase is the minimum time of 19 ns.

## **HORizontal:REF<x>:MAGnify<x>:RECORDlength? (Query Only)**

This query only command returns the record length of the Reference waveform specified by x, which can be 1 through 8, using the specified Mag timebase (1 or 2). The value of the record length can be 50, 100, 250, 500, 1000, 2000, and 4000 samples.

**Group** Horizontal

**Syntax** `HORizontal:REF<x>:MAGnify<x>:RECORDlength?`

**Related Commands** [HORizontal:REF<x>:MAGnify<x>:SCALE?](#), [HORizontal:REF<x>:MAGnify<x>:RESolution?](#)

**Examples** `HORIZONTAL:REF5:MAG2:RECORDLENGTH?` might return `HORIZONTAL:REF5:MAG2:RECORDLENGTH 1000`, indicating that the record length of the Ref 5 waveform using the Mag 2 timebase is 1000 samples.

## **HORizontal:REF<x>:MAGnify<x>:REFPoint? (Query Only)**

This query only command returns the horizontal reference point of the specified reference waveform using the specified magnification timebase. The reference waveform is specified by x, which can be 1 through 8 using the specified Mag timebase (1 or 2).

Reference waveforms have a fixed horizontal scale and are unaffected by the `HORizontal:MAGnify<x>:SCALE` command.

**Group** Horizontal

**Syntax** `HORizontal:REF<x>:MAGnify<x>:REFPoint?`

**Related Commands** [HORizontal:REF<x>:MAGnify<x>:SCALE?](#)  
[HORizontal:REF<x>:MAGnify<x>:RESolution?](#)

**Examples** `HORIZONTAL:REF5:MAG2:REFPOINT?` might return `HORIZONTAL:REF5:MAG2:REFPOINT 5.0000000000E-001`, indicating that the horizontal magnify reference point for this reference waveform is at 50% of the Mag2 window.

## **HORizontal:REF<x>:MAGnify<x>:RESolution? (Query Only)**

This query only command returns the current resolution per sample of the Reference waveform. The Reference waveform is specified by `x`, which can be 1 through 8; the Mag timebase is specified by `x`, which can be 1 or 2. Resolution is always displayed in time units.

**Group** Horizontal

**Syntax** `HORizontal:REF<x>:MAGnify<x>:RESolution?`

**Related Commands** [HORizontal:REF<x>:MAGnify<x>:SCALE?](#), [HORizontal:REF<x>:MAGnify<x>:RECORDlength?](#)

**Examples** `HORIZONTAL:REF1:MAG1:RESOLUTION?` might return `HORIZONTAL:REF1:MAGNIFY1:RESOLUTION 12.5000000000E-012`, indicating that the resolution of the Ref 1 waveform using the Mag1 timebase is 12.5 ps.

## **HORizontal:REF<x>:MAGnify<x>:SCALE? (Query Only)**

This query only command returns the scale (time per division) of the specified Reference waveform using the specified Mag timebase. The Reference waveform



is specified by x, which can be 1 through 8; the Mag timebase is specified by x, which can be 1 or 2.

**Group** Horizontal

**Syntax** `HORizontal:REF<x>:MAGnify<x>:SCALE?`

**Related Commands** [HORizontal:REF<x>:MAGnify<x>:RECOrdlength?](#), [HORizontal:REF<x>:MAGnify<x>:RESolution?](#)

**Examples** `HORIZONTAL:REF3:MAG2:SCALE?` might return `HORIZONTAL:REF3:MAGNIFY2:SCALE 1.00000000000E-010`, indicating that the Mag2 timebase is 100 ps per division for the Ref 3 waveform.

## **HORizontal:REF<x>:MAGnify<x>:TOFPoint? (Query Only)**

This query only command returns time of first point of the specified Reference waveform using the specified Mag timebase. The Reference waveform is specified by x, which can be 1 through 8; the Mag timebase is specified by x, which can be 1 or 2. The time of first point is the time interval between the trigger point and the first point in the specified Mag timebase record (waveform).

**Group** Horizontal

**Syntax** `HORizontal:REF<x>:MAGnify<x>:TOFPoint?`

**Related Commands** [HORizontal:REF<x>:MAGnify<x>:POSition?](#), [HORizontal:REF<x>:MAGnify<x>:SCALE?](#)

**Examples** `HORIZONTAL:REF1:MAG1:TOFPPOINT?` might return `HORIZONTAL:REF1:MAGNIFY1:TOFPPOINT 50.0000000000E-009`, indicating that the time interval between the trigger point and the first point in the Ref 1 waveform record using the Mag1 timebase is 50 ns.

## **HORizontal:REF<x>:MAGnify<x>:TOLPoint? (Query Only)**

This query only command returns time of last point of the specified Reference waveform using the specified Mag timebase. The Reference waveform is specified by x, which can be 1 through 8; the Mag timebase is specified by x, which can be

1 or 2. The time of last point is the time interval between the trigger point and the last point in the specified Mag timebase record (waveform).

**Group** Horizontal

**Syntax** `HORizontal:REF<x>:MAGnify<x>:TOLPoint?`

**Related Commands** [HORizontal:REF<x>:MAGnify<x>:POSition?](#), [HORizontal:REF<x>:MAGnify<x>:SCALE?](#)

**Examples** `HORIZONTAL:REF1:MAG1:TOLPOINT?` might return `HORIZONTAL:REF1:MAGNIFY1:TOLPOINT 500.0000000000E-9`, indicating that the time of last point of the Ref 1 waveform record using the Mag 1 timebase is 500 ns.

## **HORizontal:REF<x>:MAIn:POSition? (Query Only)**

This query only command returns the horizontal position of the specified reference waveform using the Main timebase. The reference waveform is specified by x, which can be 1 through 8.

**Group** Horizontal

**Syntax** `HORizontal:REF<x>:MAIn:POSition?`

**Related Commands** [HORizontal:REF<x>:MAIn:REFPoint?](#), [HORizontal:REF<x>:MAIn:TOFPoint?](#), [HORizontal:REF<x>:MAIn:TOLPoint?](#)

**Examples** `HORIZONTAL:REF5:MAIN:POSITION?` might return `HORIZONTAL:REF5:MAIN:POSITION 51.3630000000E-009`, indicating that the horizontal position of the Reference 5 waveform using the Main timebase is 51.36 ns.

## **HORizontal:REF<x>:MAIn:RECOrdlength? (Query Only)**

This query only command returns the record length of the specified reference waveform using the Main timebase. The reference waveform is specified by x, which can be 1 through 8. The value of the record length can be 50, 100, 250, 500, 1000, 2000, 4000, 8000 and 16000 samples.

**Group** Horizontal

**Syntax** `HORizontal:REF<x>:MAIn:RECOrdLength?`

**Related Commands** [HORizontal:REF<x>:MAIn:RESolution?](#), [HORizontal:REF<x>:MAIn:SCAle?](#)

**Examples** `HORIZONTAL:REF7:MAIN:RECORDLENGTH?` might return `HORIZONTAL:REF7:MAIN:RECORDLENGTH 1000`, indicating that the record length of the Reference 7 waveform using the Main timebase is 1000 samples.

## **HORizontal:REF<x>:MAIn:REFPoint? (Query Only)**

This query only command returns the horizontal reference point of the specified reference waveform using the Main timebase. The reference waveform is specified by x, which can be 1 through 8.

**Group** Horizontal

**Syntax** `HORizontal:REF<x>:MAIn:REFPoint?`

**Related Commands** [HORizontal:REF<x>:MAIn:SCAle?](#)

**Examples** `HORIZONTAL:REF1:MAIN:REFPOINT?` might return `HORIZONTAL:REF1:MAIN:REFPOINT 5.000000000000E-001`, indicating that the horizontal reference point of Reference 1 waveform using the Main timebase is set to 0.5 (50%).

## **HORizontal:REF<x>:MAIn:RESolution? (Query Only)**

This query only command returns the current resolution per sample of the specified reference waveform using the Main timebase. The reference waveform is specified by x, which can be 1 through 8. The resolution value is the same as the value displayed in the Resolution field in the Timebase section of the Horizontal Setup dialog box. The resolution is determined by the scale and recordlength values of the reference waveform. Resolution is always displayed in time units.

**Group** Horizontal

**Syntax**      `HORizontal:REF<x>:MAIn:RESolution?`

**Related Commands**      [HORizontal:REF<x>:MAIn:SCALE?](#), [HORizontal:REF<x>:MAIn:RECORDlength?](#)

**Examples**      `HORIZONTAL:REF7:MAIN:RESOLUTION?` might return  
`HORIZONTAL:REF7:MAIN:RESOLUTION 10.0000000000E-012`,  
 indicating that the Reference 7 waveform resolution using the Main timebase is 10 ps.

## **HORizontal:REF<x>:MAIn:SCALE? (Query Only)**

This query only command returns the scale (time per division) of the Main timebase for the specified reference waveform. The reference waveform is specified by x, which can be 1 through 8.

The horizontal scale that is returned with this command is unique to the specified reference waveform. The horizontal scale, together with the record length value, determines the sample interval.

**Group**      Horizontal

**Syntax**      `HORizontal:REF<x>:MAIn:SCALE?`

**Related Commands**      [HORizontal:REF<x>:MAIn:RECORDlength?](#), [HORizontal:REF<x>:MAIn:RESolution?](#)

**Examples**      `HORIZONTAL:REF3:MAIN:SCALE?` might return  
`HORIZONTAL:REF3:MAIN:SCALE 2.0000000000E-009`, indicating that the Main timebase is 2 ns per division for the Reference 3 waveform.

## **HORizontal:REF<x>:MAIn:TOFPoint? (Query Only)**

This query only command returns time of first point of the specified reference waveform record using the Main timebase. The time of first point is the time interval between the trigger point and the first point in the specified Reference Main timebase record.

**Group**      Horizontal

**Syntax**    `HORizontal:REF<x>:MAIn:TOfPoint?`

**Related Commands**    [HORizontal:REF<x>:MAIn:POStion?](#), [HORizontal:REF<x>:MAIn:REFPoint?](#), [HORizontal:REF<x>:MAIn:SCAlE?](#)

**Examples**    `HORIZONTAL:REF1:MAIN:TOfPOINT?` might return `HORIZONTAL:REF1:MAIN:TOfPOINT 41.36300000000E-009`, indicating that the time interval between the trigger point and the first point in the record for the Reference 1 Main timebase record is 41.36 ns.

## **HORizontal:REF<x>:MAIn:TOLPoint? (Query Only)**

This query only command returns time of last point of the specified reference record (waveform) using the Main timebase. The time of last point is the time interval between the trigger point and the last point in the specified Reference Main timebase record.

**Group**    Horizontal

**Syntax**    `HORizontal:REF<x>:MAIn:TOLPoint?`

**Related Commands**    [HORizontal:REF<x>:MAIn:POStion?](#), [HORizontal:REF<x>:MAIn:REFPoint?](#), [HORizontal:REF<x>:MAIn:SCAlE?](#)

**Examples**    `HORIZONTAL:REF3:MAIN:TOLPOINT?` might return `HORIZONTAL:REF3:MAIN:TOLPOINT 51.30000000000E-009`, indicating that the time interval between the trigger point and the last point in the Reference 3 Main timebase record is 51.3 ns.

## **HORizontal:UNIts**

This command sets or queries the horizontal units used for all timebases. The available units are seconds, bits, meters, feet, and inches. This is equivalent to selecting Seconds Bits, or Distance for Units in the All Timebases section of the Horizontal Setup dialog box. When selecting Distance for horizontal units, further options are meters, feet and inches.

**Group**    Horizontal

**Syntax**      `HORIZONTAL:UNITS {S | BITS | FT | IN | M}`  
`HORIZONTAL:UNITS?`

**Related Commands**    [HORIZONTAL:DISTANCE:DIELECTRIC](#)

- Arguments**
- S sets the horizontal units for all timebases to seconds.
  - BITS sets the horizontal units for all timebases to bits.
  - FT sets the horizontal units for all timebases to feet.
  - IN sets the horizontal units for all timebases to inches.
  - M sets the horizontal units for all timebases to meters.

**Examples**      `HORIZONTAL:UNITS S` sets the horizontal units for all timebases to seconds.  
`HORIZONTAL:UNITS?` might return `HORIZONTAL:UNITS BITS`, indicating that horizontal units for all timebases is set to bits.

## ID? (Query Only)

This is a query only command that returns identifying information about the instrument and its firmware.

**Group**          Status and Error

**Syntax**        `ID?`

**Related Commands**    [\\*IDN?](#)

**Returns**        The instrument id in the following format:  
`TEK/<model number>,CF:92.1CT,FV:<firmware version number>`

**Examples**        `ID?` might return `ID TEK/DSA8300,CF:91.1CT,FV:3.0.444`.

## \*IDN? (Query Only)

This is a query only command that returns the instrument identification code.

<b>Group</b>	Miscellaneous
<b>Syntax</b>	*IDN?
<b>Related Commands</b>	ID?
<b>Returns</b>	The instrument ID in the following format: TEKTRONIX,<model number>,0,CF:92.1CT FV <firmware version number>
<b>Examples</b>	*IDN? might return TEKTRONIX,DSA8300,B010112,CF:91.1CT FV:5.5.214.

## IMPort:CUSTommask (No Query Form)

This command (no query form) imports and loads the custom mask from the specified file path.

<b>Group</b>	Save and Recall
<b>Syntax</b>	IMPort:CUSTommask <file path>
<b>Arguments</b>	<FILE PATH> is a quoted string that defines the file name and path. Input the file path using the form <drive>/<dir>/<filename>. <drive> and one or more <dir>s are optional. Do not use wild card characters.
<b>Examples</b>	IMPORT:CUSTOMMASK "C:/MY Documents/TekScope/UI/Masks/Custommask1.m8k" imports and loads the custom mask named Custommask1.m8k from the Masks subdirectory on the C drive.

## LOCK

This command sets or queries the lock state of the touchscreen and the front-panel controls. There is no front-panel equivalent.

<b>Group</b>	Miscellaneous
--------------	---------------

**Syntax** LOCK {ALL | NONE | TOUCH | FPAn1}  
LOCK?

**Related Commands** UNLock

- Arguments**
- ALL disables all front-panel controls and the touchscreen.
  - NONE enables all front-panel controls and the touchscreen. This is equivalent to the UNLock ALL command.

---

**NOTE.** *If the instrument is in the Remote With Lockout State (RWLS), the LOCK NONE command has no effect. For more information, see the ANSI/IEEE Std. 488.1-1987 Standard Digital Interface for Programmable Instrumentation, section 2.8.3 on RL State Descriptions.*

---

- TOUCH disables the touchscreen.
- FPAn1 disables the front-panel controls.

**Examples** LOCK ALL locks the front-panel controls and the touchscreen.  
LOCK? might return LOCK NONE when the front-panel controls and touchscreen are enabled by this command.

## \*LRN? (Query Only)

This query only command returns a string listing the instrument settings, except for configuration information for the calibration values. You can use this string to return the instrument to the state it was in when you made the \*LRN? query. This command is identical to the SET? query.

**Group** Miscellaneous

**Syntax** \*LRN?

---

**NOTE.** *The \*LRN? query always returns a string including command headers, regardless of the setting of the HEADer command. This is because the returned string is intended to be sent back to the instrument as a command string. The VERBoSe command can still be used normally to specify whether the returned headers should be abbreviated.*

---



**Related Commands**    [HEADer](#), [SET?](#), [VERBose](#)

**Examples**    \*LRN? might return ACQUIRE:STATE 1;MODE SAMPLE; NUMENV 10;NUMAVG 16; REPET 1; STOPAFTER RUNSTOP; :DIAG:LOOP:OPTION ONCE; COUNT 1; :DIAG:STATE HALT; :HEADER 1;:VERBOSE 1; CURSOR:FUNCTION OFF; VBAR:UNITS SECONDS; POSITION1 1.00E-6; POSITION2 9.00E-6; SELECT CURSOR1.

## MASK? (Query Only)

This query only command returns the status of all settable mask parameters.

**Group**    Mask

**Syntax**    MASK?

**Examples**    MASK? might return MASK:AUTOS:MOD MAN; HIL:METH MEAN; :MASK:STAN OC12; COLO 16711935; COUN:STATE 1; TOT 617435; WAV 3811;SAMP 1905500; MASK:DIS 1;MAR:PERC 0.0000;STATE 1; MASK:MASK1:COUN 0; NR\_P 4; POI 20.69040050401E-9,0.00100000005, 20.69499949187E-9,799.99997979030E-6, 22.30039974904E-9,799.99997979030E-6, 22.30500051326E-9,0.00100000005; POINTSP14.8296,0.0000,15.0300,10.0250, 84.9699,10.0250,85.1703,0.0000; MASK:MASK2:COUN 617435;NR\_P 6; POI 21.09520025328E-9,0.0000, 21.33899990042E-9,-340.54054412991E-6, 21.65639934049E-9,-340.54054412991E-6, 21.90020076398E-9,0.0000, 21.65639934049E-9,345.94594035298E-6, 21.33899990042E-9,345.94594035298E-6; POINTSP 32.4649,50.1253,43.0861,67.1679, 56.9138,67.1679,67.5350,50.1253,56.9138, 32.8320,43.0861,32.8320; MASK:MASK3:COUN 0;NR\_P 4; POI 20.69040050401E-9,-0.00100000005, 22.30500051326E-9,-0.00100000005, 22.30039974904E-9,-799.99997979030E-6, 20.69499949187E-9,-799.99997979030E-6; POINTSP 14.8296,100.2506,85.1703,100.2506, 84.9699,90.2255,15.0300,90.2255; MASK:MASK4:COUN -1;NR\_P 0; POI 0,0; POINTSP 0,0; :MASK:MASK5:COUN -1; NR\_P 0;POI 0,0;POINTSP 0,0; :MASK:MASK6:COUN -1; NR\_P 0;POI 0,0;POINTSP 0,0; :MASK:MASK7:COUN -1; NR\_P 0;POI 0,0;POINTSP 0,0;

```
:MASK:MASK8:COUN -1; NR_P 0;POI 0,0;POINTSP 0,0; :MASK:SOU
CH1,MAIN; WFMDB:STATE 1.
```

## MASK:AUTOSet:HILOW:METhod

This command sets or queries the method used for determining the High value and Low value used when aligning the input signal to NRZ eye masks. The method set will be used when Autoset executes on a mask source (that is, when a mask source is the currently selected waveform). The query form of this command returns the current method set for Mask Autoset.

The default Mask Autoset High/Low method is MEAN. The use of this command is equivalent to setting the HiLow Method in the Mask Autoset section of the Mask Setup dialog box.

**Group** Mask

**Syntax** MASK:AUTOSet:HILOW:METhod {MEAN | MODE}  
 MASK:AUTOSet:HILOW:METhod?

**Related Commands** [AUTOSet](#), [MASK:AUTOSet:MODE](#), [MASK:STANdard](#)

- Arguments**
- MEAN sets Mask Autoset to use the mean of the High level (topline) and Low level (baseline) within a fixed eye aperture (center 20% of the eye) to align the input signal to the NRZ mask. Mean is the default state and is automatically selected whenever a mask communication standard other than User or none is specified.
  - MODE sets Mask Autoset to use the mode of the High level (topline) and Low level (baseline) across one unit interval of the eye diagram to align the input signal to the NRZ mask. Setting the method to Mode coerces the mask communication standard to User.

**Examples** MASK:AUTOSET:HILOW:METHOD MODE sets the Mask Autoset high/low method to Mode.

MASK:AUTOSET:HILOW:METHOD? might return MEAN, indicating that the current Mask Autoset High/Low method setting is Mean.

## MASK:AUTOSet:MODE

This command sets or queries whether or not a mask autoselect will automatically be done after a standard mask is selected. This is equivalent to selecting Automatic or Manual in the Autoselect section of the Mask Setup dialog box.

**Group** Mask

**Syntax** MASK:AUTOSet:MODE {AUTO | MANua1}  
MASK:AUTOSet:MODE?

- Arguments**
- **AUTO** sets the instrument to automatically run the mask autoselect when a standard mask (or a user-defined mask that is derived from a standard mask) is selected from the Comm Standard section of the Mask Setup dialog box. Autoselect will not run continuously; it will run once when a standard is selected.
  - **MANua1** sets the mask autoselect algorithm for a standard mask to run only if the user presses the autoselect button or sends the **AUTOSet EXECute** command. If a standard mask (or a user-defined mask that is derived from a standard mask) is not currently selected, the mask autoselect algorithm will not run, and the normal autoselect will run when the user presses the Autoselect button. Manual is the default value.

**Examples** MASK:AUTOSET:MODE AUTO sets the instrument to automatically run autoselect when a standard mask is selected from the Comm Standard section of the Mask Setup dialog box.

MASK:AUTOSET:MODE? might return MASK:AUTOSET:MODE AUTO, indicating that mask autoselect will run automatically when the user selects a standard mask from the front panel.

## MASK:COLOr

This command sets or queries the color of the mask. (The command is the same as the DISplay:COLor:MASK command.) This is equivalent to setting Color in the Options section of the Mask Setup dialog box. The default color for mask polygons is fuchsia.

The argument for the command is an integer equal to the sum of three color values (red, green, and blue). You specify the value for each of the three color values and sum them to specify the color.

The following table lists values for common colors.

### Index number values for common colors

Name	Index number
Black	0
Blue	16711680
Aqua	16776960
Lime	65280
Fuchsia	16711935
Red	255
Yellow	65535
White	16777215
Navy	8388608
Teal	8421376
Green	32768
Purple	8388736
Maroon	128
Olive	32896
Gray	8421504
Silver	12632193

**Group** Mask

**Syntax** MASK:COLOr <NR1>  
MASK:COLOr?

**Related Commands** [DISplay:COLor:MASK](#)

**Arguments** NR1 is an integer in the range of 0 through 16777215 representing a color defined as follows:  $\text{red} + 256 \times \text{green} + 65536 \times \text{blue}$  where red, green and blue are values between 0 and 255.

**Examples** MASK:COLOR 16711935 sets the color of the mask to fuchsia.  
MASK:COLOR? might return MASK:COLOR 32768, indicating that the color of the mask is green.

## MASK:COUNT

This command clears the counts for the masks, clears the data source, and restarts the counting. This is equivalent to clicking Clear in the Mask Counts section

of the Mask Setup dialog box. The query form of this command returns all the values for the mask count parameters.

**Group** Mask

**Syntax** MASK:COUNT {RESET}  
MASK:COUNT?

**Examples** MASK:COUNT RESET clears the counts for the masks, clears the data source, and restarts the counting.

MASK:COUNT? might return MASK:COUNT:STATE 1;TOTAL 487641;WAVEFORMS 2887;SAMPLES 1443500.

## MASK:COUNT:SAMPLES? (Query Only)

This query only command returns the total number of sample points used in the count of the total number of mask hits (returned by the MASK:COUNT:TOTAL? query).

**Group** Mask

**Syntax** MASK:COUNT:SAMPLES?

**Examples** MASK:COUNT:SAMPLES? might return MASK:COUNT:SAMPLES 500, indicating that the count of mask hits is based on 500 samples.

## MASK:COUNT:STATE

This command sets or queries whether or not mask counting is being done. Sending this command is equivalent to checking or clearing the Enable Mask Counts check box in the Source section of the Mask Setup dialog box.

**Group** Mask

**Syntax** MASK:COUNT:STATE {ON | OFF | <NR1>}  
MASK:COUNT:STATE?

- Arguments**
- OFF turns off mask counting. This is the default state.
  - ON turns on mask counting.
  - NR1 set to 0 turns off mask counting; any other integer turns on mask counting.

**Examples** MASK:COUNT:STATE ON turns on mask counting.

MASK:COUNT:STATE? might return MASK:COUNT:STATE 0, indicating that mask counting is turned off.

---

**NOTE.** *A query always returns a 0 or 1.*

---

## MASK:COUNT:TOTAL? (Query Only)

This query only command returns the sum of all the hits in all the defined masks. If masks overlap (no standard masks overlap) and a hit occurs in the overlap region, the total hits will count that hit multiple times. The query returns the current value, which may be displayed in the Mask readout as Total.

**Group** Mask

**Syntax** MASK:COUNT:TOTAL?

**Returns** NR1

**Examples** MASK:COUNT:TOTAL? might return MASK:COUNT:TOTAL 20057, indicating that the total hits in all masks is 20,057.

## MASK:COUNT:WAVEforms? (Query Only)

This query only command returns the number of waveforms that have contributed to mask counting. This value may be displayed in the Mask readout as Waveforms.

**Group** Mask

**Syntax** MASK:COUNT:WAVEforms?

**Returns** NR1

**Examples** MASK:COUNT:WAVEFORMS? might return MASK:COUNT:WAVEFORMS 6568, indicating that the total number of waveforms contributed to mask counting is 6,568.

## MASK:DISplay

This command sets or queries whether or not defined masks are displayed on the screen. This is useful for temporarily turning off user-defined masks without deleting them. This is equivalent to checking or clearing the Display Mask control in the Options section of the Mask Setup dialog box.

**Group** Mask

**Syntax** MASK:DISplay {OFF | ON | <NR1>}  
MASK:DISplay

- Arguments**
- OFF removes the masks from the display; this does not turn mask counting off. This is the default value.
  - ON displays the masks. Turning mask counting on turns this on. Selecting any standard mask turns this on. MASK:MASK<x>:POINTSPCNT or MASK:MASK<x>:POINTS turns this on.
  - NR1 set to 0 turns off the mask display; any other integer turns on the mask display.

**Examples** MASK:DISPLAY ON sets the display to show the defined masks.  
MASK:DISPLAY? might return 1, indicating that the display shows masks.

---

**NOTE.** A query always returns a 0 or 1.

---

## MASK:MARgin:PERCent

This command sets or queries the mask-margin percentage. This is equivalent to setting a value in the Margin control in the Mask Margins section of the Mask Setup dialog box. If the margin percentage would cause the masks to either go off screen or cross one another, the margin is not allowed, and the last valid margin is restored. If you set a positive mask margin percentage, the polygon expands and the mask test will be more difficult to pass due to the expanded test area. If you set a negative mask margin percentage, the polygon shrinks and the mask test will be easier to pass due to the smaller area.

<b>Group</b>	Mask
<b>Syntax</b>	MASK:MARGin:PERCent <NR3> MASK:MARGin:PERCent?
<b>Arguments</b>	NR3 value is -90.0 to +100.0. The default is 5.
<b>Examples</b>	MASK:MARGIN:PERCENT 25 sets the mask margin percentage to 25. MASK:MARGIN:PERCENT? might return MASK:MARGIN:PERCENT 1.000000000000E+001, indicating that the mask margin is set to 10%.

## MASK:MARGin:STATE

This command sets or queries the state of the mask margins. This is equivalent to checking or clearing the On check box in the Mask Margins section of the Mask Setup dialog box.

<b>Group</b>	Mask
<b>Syntax</b>	MASK:MARGin:STATE {ON   OFF   <NR1>} MASK:MARGin:STATE?
<b>Arguments</b>	<ul style="list-style-type: none"> <li>■ OFF turns off mask margins. The default is OFF. The currently displayed margined mask is erased and the original mask is displayed.</li> <li>■ ON turns on mask margins.</li> <li>■ NR1 set to 0 turns off mask margins; any other integer turns on the selected mask margins.</li> </ul>
<b>Examples</b>	MASK:MARGIN:STATE ON turns on mask margins. MASK:MARGIN:STATE? might return MASK:MARGIN:STATE 1, indicating that mask margins are on.

---

**NOTE.** A query always returns a 0 or 1.

---



**MASK:MASK<x>**

This command deletes all points in the mask specified by x, which can be 1 through 8. The query form of this command returns all the parameters for the specified mask.

<b>Group</b>	Mask
<b>Syntax</b>	MASK:MASK<x> {DELEte} MASK:MASK<x>?
<b>Arguments</b>	DELEte deletes all points in the specified mask.
<b>Examples</b>	<p>MASK:MASK3 DELETE deletes the points in Mask 3.</p> <p>MASK:MASK2? might return MASK:MASK2:COUNT 973938; NR_PT 6; POINTS 2.41362680953E-008,1.25313281387E-005, 2.80024003985E-008, -1.59147870727E-003, 3.37463674782E-008, -1.59147870727E-003, 3.76125015578E-008,1.25313281387E-005, 3.37463674782E-008,1.61654129624E-003, 2.80024003985E-008,1.61654129624E-003; POINTSPCNT 25.4000,49.7500,39.4000,65.7500, 60.2000,65.7500,74.2000,49.7500,60.2000, 33.7500,39.4000,33.7500.</p>

**MASK:MASK<x>:COUNT? (Query Only)**

This query only command returns the number of hits in the specified mask. The mask is specified by <x>, which can be 1 through 8. The count will be zero unless the MASK:COUNT:STATE is ON (or was ON).

<b>Group</b>	Mask
<b>Syntax</b>	MASK:MASK<x>:COUNT?
<b>Related Commands</b>	<a href="#">MASK:COUNT:STATE</a>
<b>Returns</b>	NR1 is the number of hits in the specified mask.

**Examples** MASK:MASK3:COUNT? might return MASK:MASK3:COUNT 1478, indicating that the number of hits in Mask 3 is 1478.

## MASK:MASK<x>:NR\_Pt? (Query Only)

This query only command returns the number of points (vertices) in the specified mask. The mask is specified by <x>, which can be 1 through 8. One use of this query is to see how many pairs a MASK:MASK<x>:POINTSPcnt or MASK:MASK<x>:POInts command will return.

**Group** Mask

**Syntax** MASK:MASK<x>:NR\_Pt?

**Related Commands** [MASK:MASK<x>:POInts](#), [MASK:MASK<x>:POINTSPcnt](#)

**Returns** <NR1> is the number of points in the specified mask.

**Examples** MASK:MASK3:NR\_PT? might return MASK:MASK3:NR\_PT 6, indicating that Mask 3 has 6 points.

## MASK:MASK<x>:POInts

This command sets or queries the points (vertices) in the specified mask in waveform coordinates. The mask is specified by <x>, which can be 1 through 8. When you send this command, any currently existing points in the mask are deleted. Each point or vertex is an xy coordinate that represents a horizontal/vertical pair in source waveform (user) units. The order of the pairs has no effect on the mask created.

If the vertical or horizontal scale or position changes after this command is executed, a query will return different values for the vertices. This is because the mask polygons remain stationary on the display, but the relationship of the internal coordinate system to the absolute user values changes.

---

**NOTE.** *If the specified mask is undefined, a query will return 0,0.*

---

**Group** Mask

<b>Syntax</b>	MASK:MASK<x>:POInts <NR3>,<NR3>,<NR3>,<NR3>, [..., ...] MASK:MASK<x>:POInts?
<b>Arguments</b>	NR3 is the pair of xy coordinates for a point (vertex) in the mask. Each mask polygon can have a maximum of 50 points or vertices. You must specify more than two pair of points, otherwise the mask is marked as undefined, and the points are ignored. The default is no points in the mask.
<b>Examples</b>	MASK:MASK7:POINTS -2.3E-9,44E-3,-2.5E-9,47E-3,1.2E-9,40E-3 defines the points in Mask 7.  MASK:MASK3:POINTS? might return MASK:MASK3:POINTS 1.97664729029E-008, -5.00000000000E-001, 2.13889173040E-008, -5.00000000000E-001, 2.13843094343E-008, -3.74686717987E-001, 1.97710825489E-008, -3.74686717987E-001, indicating the points in Mask 3.

## MASK:MASK<x>:POINTSPcnt

This command sets or queries the points (vertices) in the specified mask in percentage coordinates. The mask is specified by <x>, which can be 1 through 8. Any currently existing points in the mask are deleted. The upper leftmost point is 0,0, and the lower rightmost point is 100,100.

Each mask polygon can have a maximum of 50 points or vertices. If more than 50 points are specified, an error is generated and the extra points are ignored. If any point is outside of 0-100, an error is generated, and the point is clipped to the nearest boundary. The order of the pairs has no effect on the mask created.

**NOTE.** *If the specified mask is undefined, a query will return 0,0.*

<b>Group</b>	Mask
<b>Syntax</b>	MASK:MASK<x>:POINTSPcnt <NR3>,<NR3>[,<NR3>,<NR3>, ...] MASK:MASK<x>:POINTSPcnt?
<b>Arguments</b>	NR3 is the pair of xy coordinates for a point (vertex) in the mask. You must specify more than two pair of points, otherwise the mask is marked as undefined, and the points are ignored. Resolution of the NR3 value is to the nearest pixel. 1000 pixels horizontally implies 0.1% resolution; 400 pixels vertically implies 0.25% resolution. The default is no points in the mask.

- Examples** MASK:MASK7:POINTSPCNT 20.4,10.5,90,10.5,50,80 defines the points in Mask 7.
- MASK:MASK7:POINTSPCNT? might return MASK:MASK7:POINTSPCNT 20.4000,10.5000,90.000,10.500,50.000,80.000 as the percentage coordinates for Mask 7.

## MASK:SOURce

This command sets or queries which source will be compared against the mask(s) when counting is turned on. This command can optionally set the timebase of the source. This is equivalent to selecting a waveform (and timebase) in the Source section of the Mask Setup dialog box. If you do not specify the timebase, the default is Main.

**Group** Mask

**Syntax** MASK:SOURce {CH(x) | MATH(x) | REF(x)}[, {MAIn | MAG1 | MAG2}]  
 MASK:SOURce?

- Arguments**
- CH<x> selects a channel waveform to be compared against the specified mask. The range for x is 1 through 8.
  - MATH<x> selects a math waveform to be compared against the specified mask. The range for x is 1 through 8.
  - REF<x> selects a reference waveform to be compared against the specified mask. The range for x is 1 through 8.
  - MAIn (optional) selects Main as the timebase for the specified waveform to be compared against the mask. This is the default timebase.
  - MAG1 (optional) selects MAG1 as the timebase for the specified waveform to be compared against the mask.
  - MAG2 (optional) selects MAG2 as the timebase for the specified waveform to be compared against the mask.

- Examples** MASK:SOURCE CH1 enables CH1 to be compared against the mask. Since a timebase is not specified, the default, Main, is used.
- MASK:SOURCE? might return MASK:SOURCE CH1,MAIN, indicating that the Channel 1 waveform will be compared against the mask using the Main timebase.

## MASK:STANdard

This command selects a set of predefined mask polygons (hereafter referred to as a “mask”) to be used for mask testing in accordance with the communications standard specified by the command argument. The query form of this command returns the current mask. Note the following usage rules:

- Setting the MASK:STANdard sets the HORizontal:BITS:STANdard to the same argument, thereby setting the horizontal bit rate to the bit rate associated with the selected standard.
- Do not use the USERMask to select a user-defined mask; rather use the MASK:MASK<X>:POINts and/or the MASK:MASK<X>:POINTSPcnt commands to define the mask. This action coerces the MASK:STANdard and HORizontal:BITS:STANdard settings to USERMask. (The USERMask setting cannot be selected directly with the MASK:STANdard command, but MASK:STANdard queries will return USERMask when user-defined masks are selected.)
- When you use user-defined masks, you should set the horizontal bit rate explicitly with the HORizontal:BITS:BITRate command.
- You can use the MASK:MASK<X> commands to access the current masks, whether predefined or user defined, and their associated counters.
- If you specify user-defined masks, note that selecting one of the predefined mask standards will overwrite your user-defined masks. If your test is related to one of the predefined mask standards, you might want to first select the related predefined standard, and then only modify the masks and/or bit rate as required.
- Specifying a mask communication standard other than User or None coerces the MASK:AUTOSet:HILow:METHod to MEAN.
- Changing the MASK:AUTOSet:HILow:METHod from MEAN to MODe coerces the mask communication standard to USER.
- Selecting NONE will delete all currently defined masks.
- If MASK:COUNt:STATE is ON, then mask counting starts.
- Custom mask definitions must be imported before it can be used.

**Group** Mask

**Syntax** MASK:STANdard { NONE | CUSTom | ATARXG1 | ATARXG2 | ATARXG3 | ATATXG1 | ATATXG2 | ATATXG3 | ENET40GB\_LR4 | ENET40GB\_SR4 | ENET1250 | ENET2500 | ENET3125 | ENET9953 | ENET10313 | ENET10GB\_LRM | ENET100B\_BX10 | ENET100GB\_ER4 | ENET100GB\_LR4 | ENET100GB\_SR10 | ENET100B\_LX10 | ENET1000B\_KX | ENET10313 | ENET11096 | ENET41250 | FC133 | FC133E | FC266 | FC266E

| FC531 | FC531E | FC1063 | FC1063E | FC2125 | FC2125E\_ABR  
 | FC2125E\_ABT | FC2125E\_AGR | FC2125E\_AGT | FC4250E\_ABR  
 | FC4250E\_ABT | FC4250E\_AGR | FC4250E\_AGT | FC8500E\_ABR  
 | FC8500E\_ABT | FC8500E\_AGR | FC8500E\_AGT | FC8500D |  
 FC8500FINAL | FC4250 | FC10519 | FC11317 | FC14025\_MMR6\_1 |  
 FC14025\_SMR6\_1 | FEC2666 | FEC10664 | FEC10709 | FEC42657  
 | FEC43018 | INF2500 | INFIE25 | INFIniband | OC1 | OC3  
 | OC9 | OC12 | OC18 | OC24 | OC36 | OC48 | OC192 | OC768  
 | PCIEXPRESS\_Rcv | PCIEXPRESS50\_Rcv | RIO\_SERIAL1G |  
 RIO\_SERIAL2G | RIO\_SERIAL3G | SAS3\_0\_XR | SAS3\_0\_XR\_AASJ  
 | SAS3\_0\_SATA | USERMask | XFI9950\_TAA | XFI9950\_RAD |  
 XFI9950\_THB | XFI9950\_RHC | XFI9950\_TMBP | XFI9950\_RMCP  
 | XAUIFar | XAUIRFar | XAUINear | XAUIRNear }

MASK:STANDARD?

### Arguments

- NONE disables the communications standards.
- CUSTOM selects the bit rate and mask associated with an imported custom mask.
- ATARXG1 selects the Serial ATA, G1 Rx, 1.5 Gb/s bit rate and mask.
- ATARXG2 selects the Serial ATA, G2 Rx, 3 Gb/s bit rate and mask.
- ATARXG3 selects the Serial ATA, G2 Rx, 6 Gb/s bit rate and mask.
- ATATXG1 selects the Serial ATA, G1 Tx, 1.5 Gb/s bit rate and mask.
- ATATXG2 selects the Serial ATA, G2 Tx, 3 Gb/s bit rate and mask.
- ATATXG3 selects the Serial ATA, G2 Tx, 6 Gb/s bit rate and mask.
- ENET40GB\_LR4 selects the 10.3125 Gb/s bit rate and mask, in accordance with the Ethernet 40GBASE-LR4 standard.
- ENET40GB\_SR4 selects the 10.3125 Gb/s bit rate and mask, in accordance with the Ethernet 40GBASE-SR4 standard.
- ENET100GB\_ER4 selects the 25.78125 Gb/s bit rate and mask, in accordance with the Ethernet 100GBASE-ER4 standard.
- ENET100GB\_LR4 selects the 25.78125 Gb/s bit rate and mask, in accordance with the Ethernet 100GBASE-LR4 standard.
- ENET100GB\_SR10 selects the 10.3125 Gb/s bit rate and mask, in accordance with the Ethernet 100GBASE-SR10 standard.
- ENET1250 selects the 1.250 Gb/s bit rate and mask, in accordance with the IEEE 802.3 standard.
- ENET2500 selects the 2.500 Gb/s bit rate (2x Gigabit Ethernet) and mask.
- ENET3125 selects the 3.125 Gb/s bit rate and mask, in accordance with the IEEE 802.3ae-2002 standard.

- ENET9953 selects the 9.95328 Gb/s bit rate and mask, in accordance with the IEEE 802.3ae-2002 standard.
- ENET10313 selects the 10.3125 Gb/s bit rate and mask, in accordance with the IEEE 802.3ae-2002 standard.
- ENET11096 selects the 11.095728 Gb/s bit rate and mask in accordance with IEEE 802.3ae-2002.
- ENET10GB\_LRM selects the 10.3125 Gb/s bit rate and mask, in accordance with IEEE 802.3aq - 2006.
- ENET100B\_BX10 selects the 125 Mb/s bit rate and mask, in accordance with IEEE 802.3ah - 2004.
- ENET100B\_LX10 selects the 125 Mb/s bit rate and mask, in accordance with IEEE 802.3ah - 2004.
- ENET1000B\_KX selects the 1.25 Gb/s bit rate and mask, in accordance with IEEE 802.3ap Draft 3.3.
- ENET41250 selects the 41.2500 Gb/s bit rate and mask, in accordance with Ethernet 40GBASE-FR standard.
- FC133 selects the 132.81 Mb/s bit rate and mask, in accordance with the ANSI X3.230-1994 standard.
- FC133E selects the 132.81 Mb/s bit rate and mask, in accordance with the ANSI X3.230-1994 standard.
- FC266 selects the 265.6 Mb/s bit rate and mask, in accordance with the ANSI X3.230-1994 standard.
- FC266E selects the 265.6 Mb/s bit rate and mask, in accordance with the ANSI X3.230-1994 standard.
- FC531 selects the 531.2 Mb/s bit rate and mask, in accordance with the ANSI X3.230-1994 standard.
- FC531E selects the 531.2 Mb/s bit rate and mask, in accordance with the ANSI X3.230-1994 standard.
- FC1063 selects the 1.0625 Gb/s bit rate and mask, in accordance with the ANSI X3.230-1994 standard.
- FC1063E selects the 1.0625 Gb/s bit rate and mask, in accordance with the ANSI X3.230-1994 standard.
- FC2125 selects the 2.125 Gb/s bit rate and mask, in accordance with the ANSI Fibre Channel Physical Interface (FC-PI) Rev 11 draft standard.
- FC2125E\_ABR selects the 2.125 Gb/s bit rate and mask, in accordance with ANSI X3.230-1999 NCITS 1235D/Rev 11, Abs, Beta, Recv.

- FC2125E\_ABT selects the 2.125 Gb/s bit rate and mask, in accordance with ANSI X3.230-1999 NCITS 1235D/Rev 11, Abs, Beta, Transm.
- FC2125E\_AGR selects the 2.125 Gb/s bit rate and mask, in accordance with ANSI X3.230-1999 NCITS 1235D/Rev 11, Abs, Gamma, Recv.
- FC2125E\_AGT selects the 2.125 Gb/s bit rate and mask, in accordance with ANSI X3.230-1999 NCITS 1235D/Rev 11, Abs, Gamma, Transm.
- FC4250 selects the 4.250 Gb/s bit rate and mask, in accordance with the ANSI Fibre Channel Physical Interface (FC-PI) Rev 11 draft standard.
- FC4250E\_ABR selects the 4.25 Gb/s bit rate and mask, in accordance with FIBRE CHANNEL PHYSICAL INTERFACES (FC-PI-2), Rev 3.0 9/13/02 Abs, Beta, Recv.
- FC4250E\_ABT selects the 4.25 Gb/s bit rate and mask, in accordance with FIBRE CHANNEL PHYSICAL INTERFACES (FC-PI-2), Rev 3.0 9/13/02 Abs, Beta, Transm.
- FC4250E\_AGR selects the 4.25 Gb/s bit rate and mask, in accordance with FIBRE CHANNEL PHYSICAL INTERFACES (FC-PI-2), Rev 3.0 9/13/02, Abs, Gamma, Recv.
- FC4250E\_AGT selects the 4.25 Gb/s bit rate and mask, in accordance with FIBRE CHANNEL PHYSICAL INTERFACES (FC-PI-2), Rev 3.0 9/13/02, Abs, Gamma, Transm.
- FC8500E\_ABR selects the 8.5 Gb/s bit rate and mask, in accordance with FIBRE CHANNEL PHYSICAL INTERFACES (FC-PI-2), Rev 3.0 9/13/02 Abs, Beta, Recv.
- FC8500E\_ABT selects the 8.5 Gb/s bit rate and mask, in accordance with FIBRE CHANNEL PHYSICAL INTERFACES (FC-PI-2), Rev 3.0 9/13/02, Abs, Beta, Transm.
- FC8500E\_AGR selects the 8.5 Gb/s bit rate and mask, in accordance with FIBRE CHANNEL PHYSICAL INTERFACES (FC-PI-2), Rev 3.0 9/13/02, Abs, Gamma, Recv.
- FC8500E\_AGT selects the 8.5 Gb/s bit rate and mask, in accordance with FIBRE CHANNEL PHYSICAL INTERFACES (FC-PI-2), Rev 3.0 9/13/02, Abs, Gamma, Transm.
- FC8500D selects the draft 8.5000 Gb/s optical bit rate and mask per T11 January 2006 proposal.
- FC8500FINAL selects the draft 8.5000 Gb/s optical bit rate and mask per Fibre Channel Optical 8GFC.
- FC10519 selects the 10.518750 Gb/s bit rate and mask, in accordance with the proposed 10 Gigabit Fibre Channel (10GFC) standard.



- FC8GFCREV6 selects the 8.5 Gb/s bit rate and mask, in accordance with Fibre Channel Physical Interfaces FC-PI-4, REV 6.00.
- FC10519 selects the 10.51875 Gb/s bit rate and mask, in accordance with the Fibre Channel Optical 10GFC standard.
- FC11317 selects the 11.317 Gb/s bit rate and mask, in accordance with the Fibre Channel Optical FC FEC standard.
- FC14025\_MMR6\_1 selects the 14.02500 Gb/s bit rate and mask, in accordance with the Fibre Channel Optical 16GFC MM r6.1 standard.
- FC14025\_SMR6\_1 selects the 14.02500 Gb/s bit rate and mask, in accordance with the Fibre Channel Optical 16GFC SM r6.1 standard.
- FEC2666 selects the 2.6660571 Gb/s bit rate and mask, in accordance with the ITU-T G.975 Recommendation.
- FEC10664 selects the 10.6642 Gb/s bit rate and mask, in accordance with the ITU-T G.975 Recommendation.
- FEC10709 selects the 10.709225 Gb/s bit rate and mask, in accordance with the ITU-T G.709 Draft Recommendation.
- FEC42657 selects the 42.6569 Gb/s bit rate and mask, in accordance with the ITU-T G.975 Recommendation.
- FEC43018 selects the 43.018414 Gb/s bit rate and mask, in accordance with the ITU-T G.709 Draft Recommendation.
- INF2500 selects the 2.500 Gb/s bit rate and mask, in accordance with the Infiniband Optical standard. Replaces the INFINIBAND argument. If INFINIBAND is set and a query is sent, the instrument returns INF2500.
- INF2500 selects the 2.500 Gb/s bit rate and mask, in accordance with the InfiniBand standard. Replaces the INFINIBAND argument. If INFINIBAND is set and a query is sent, the instrument returns INF2500.
- INFIE25 selects the 2.5 Gb/s bit rate and mask, in accordance with the InfiniBand, IBTA Spec 1.0a, 2.5 Electrical, 2.5 Gb/s standard.
- OC1 selects the 51.84 Mb/s bit rate and mask, in accordance with the Sonet/SDH OC-1/STM-0 standard.
- OC3 selects the 155.52 Mb/s bit rate and mask, in accordance with the Sonet/SDH OC-3/STM-1 standard.
- OC9 selects the 466.56 Mb/s bit rate and mask, in accordance with the Sonet/SDH OC-9/STM-3 standard.
- OC12 selects the 622.08 Mb/s bit rate and mask, in accordance with the Sonet/SDH OC-12/STM-4 standard.
- OC18 selects the 933.12 Mb/s bit rate and mask, in accordance with the Sonet/SDH OC-18/STM-6 standard.

- OC24 selects the 1.24416 Gb/s bit rate and mask, in accordance with the Sonet/SDH OC-24/STM-8 standard.
- OC36 selects the 1.86624 Gb/s bit rate and mask, in accordance with the Sonet/SDH OC-36/STM-12 standard.
- OC48 selects the 2.48832 Gb/s bit rate and mask, in accordance with the Sonet/SDH OC-48/STM-16 standard.
- OC192 selects the 9.95328 Gb/s bit rate and mask, in accordance with the Sonet/SDH OC192/STM-64 standard.
- OC768 selects the 39.8131 Gb/s bit rate and mask, in accordance with the Sonet/SDH OC-768/STM-256 standard.
- PCIEXPRESS\_RCV selects the 2.5 Gb/s bit rate and mask, in accordance with the PCI Express Receiver, 2.5 Gb/s standard.
- PCIEXPRESS50\_RCV selects the 5 Gb/s bit rate and mask, in accordance with the PCI Express Receiver, 5 Gb/s standard.
- RIO\_SERIAL1G selects the 1.25 Gb/s bit rate and mask, in accordance with the Rapid IO Serial, 1.25 Gb/s standard.
- RIO\_SERIAL2G selects the 2.5 Gb/s bit rate and mask, in accordance with the Rapid IO Serial, 2.5 Gb/s standard.
- RIO\_SERIAL3G selects the 3.25 Gb/s bit rate and mask, in accordance with the Rapid IO Serial, 3.25 Gb/s standard.
- SAS3\_0\_XR selects the 3 Gb/s bit rate and mask.
- SAS3\_0\_XR\_AASJ selects the 3 Gb/s applied sinusoidal jitter bit rate and mask.
- SAS3\_0\_SATA selects the 3 Gb/s bit rate and mask.
- USERMask selects user-defined masks for testing (see Description, above).
- XFI9950\_TAA selects the 9.95 Gb/s bit rate and mask.
- XFI9950\_RAD selects the 9.95 Gb/s bit rate and mask.
- XFI9950\_THB selects the 9.95 Gb/s bit rate and mask.
- XFI9950\_RHC selects the 9.95 Gb/s bit rate and mask.
- XFI9950\_TMBP selects the 9.95 Gb/s bit rate and mask.
- XFI9950\_RMCP selects the 9.95 Gb/s bit rate and mask.
- XAUIFar selects the 3.1250 Gb/s far-end normalized bit rate and mask, in accordance with the IEEE 802.3ae-2002 standard.
- XAUIRFar selects the 3.1250 Gb/s far-end relative bit rate and mask, in accordance with the IEEE 802.3ae-2002 standard.

- XAUINear selects the 3.1250 Gb/s near-end normalized bit rate and mask, in accordance with the IEEE 802.3ae-2002 standard.
- XAUIRNear selects the 3.1250 Gb/s near-end relative bit rate and mask, in accordance with the IEEE 802.3ae-2002 standard.

**Examples** MASK:STANDARD OC3 selects the Sonet/SDH OC-3/STM-1 mask standard with a data rate of 155.52 Mb/s.

MASK:STANDARD? might return MASK:STANDARD OC12, indicating that the Sonet/SDH OC-12/STM-4 optical mask with a data rate of 622.08 Mb/s is selected as the current mask standard.

## MASK:WFMDB:STATE

This command sets or queries whether the MASK counting is made on a waveform database. This is equivalent to selecting or clearing the Use Wfm Database check box in the Source section of the Mask Setup dialog box.

**Group** Mask

**Syntax** MASK:WFMDB:STATE { ON | OFF | NR1 }  
MASK:WFMDB:STATE?

- Arguments**
- ON enables mask counting on a waveform database.
  - OFF disables mask counting on a waveform database (mask counting is then performed on directly on the acquired waveform).
  - NR1 set to 0 disables the mask counting on a waveform database; any other value enables mask counting on a waveform database.

**Examples** MASK:WFMDB:STATE ON enables mask counting on a waveform database.

MASK:WFMDB:STATE? might return MASK:WFMDB:STATE 0, indicating that the mask counting on a waveform database is disabled.

---

**NOTE.** A query always returns a 0 or 1.

---

## MATH<x>? (Query Only)

This is a query only command that returns the settings for the math waveform specified by <x>, which can be 1 through 8.

<b>Group</b>	Math
<b>Syntax</b>	MATH<x>?
<b>Examples</b>	MATH3? might return MATH3:POSITION 1.5940000000+001; SCALE 1.0000000000-001;DEFINE "C3+C3/C3";UNITS "v"; FILTER:RISETIME 1.0000000000-009;MODE CENTERED; MATH3NUMAVG 2.

## MATH<x>:DEFine

This command allows you to define new waveforms using mathematical expressions. Sending this command is equivalent to selecting a math waveform (M1 through M8) and entering a math expression in the Define Math dialog box. The query form of this command returns the math definition for the specified math waveform.

You can specify a math expression from waveforms, measurements and scalar sources, functions, operators, and numerical constants. You can define and display up to eight math waveforms simultaneously.

Math expressions can be simple, such as C1, which specifies that a waveform should show the signal source of Channel 1 with no mathematical computation. Math expressions can also be complex, consisting of 100 plus characters and comprising many sources, functions, and operands. As an example, you can enter the expression  $\text{Log}(C1+C2)$ , which specifies that the signals from Channels 1 and 2 are to be algebraically added, and the base 10 log of the sum is to be shown as the final math waveform.

For detailed information about constructing mathematical expressions, refer to the online help for this instrument.

<b>Group</b>	Math
<b>Syntax</b>	MATH<x>:DEFine <QString> MATH<x>:DEFine?
<b>Related Commands</b>	<a href="#">SElect:MATH&lt;x&gt;</a>
<b>Arguments</b>	QString is the mathematical expression that defines the waveform.

**Examples** `MATH2:DEFINE "C1+C2"` adds the Channel 1 waveform and Channel 2 waveform, and stores the result in Math 2.

`MATH1:DEFINE?` might return `MATH1:DEFINE "C2*R2"` as the expression that defines Math 1.

## MATH<x>:FILTER:MODE

This command sets or queries the filter mode for the math waveform specified by x, which can be 1 through 8.

**Group** Math

**Syntax** `MATH<x>:FILTER:MODE {CENTERed | SHIFted}`  
`MATH<x>:FILTER:MODE?`

**Related Commands** [MATH<x>:DEFine](#), [MATH<x>:FILTER:RISetime](#)

- Arguments**
- `CENTERed` invokes a centered filter mode that is noncausal and uses an average that is centered on each point. This is the default.
  - `SHIFted` invokes a shifted filter mode that is causal and forms the average for each point based only on preceding points.

**Examples** `MATH2:FILTER:MODE CENTERED` sets the filter mode to centered for the specified math waveform.

`MATH5:FILTER:MODE?` might return `MATH5:FILTER:MODE SHIFTED`, indicating that the shifted filter mode is applied to the Math 5 waveform.

## MATH<x>:FILTER:RISetime

This command sets or queries the risetime (bandwidth) of the math filter function.

**Group** Math

**Syntax** `MATH<x>:FILTER:RISetime <NR3>`  
`MATH<x>:FILTER:RISetime?`

**Related Commands** [MATH<x>:DEFine](#), [MATH<x>:FILTER:MODE](#)

**Arguments** NR3 is the value of the risetime in the range 1 ps to 10 s. Given the characteristics of the DSA8300 and available sampling modules, reasonable values range from ~10 ps to 100 ns.

**Examples** MATH2:FILTER:RISETIME 1.0E-12 sets the risetime of the Math 2 risetime filter to 1 ps.

MATH2:FILTER:RISETIME? might return MATH2:FILTER:RISETIME 1.000000000000E-009, indicating that the risetime of the math filter used for the Math 2 waveform is set to 1 ns.

## MATH<x>:NUMavg

This command sets or queries the number of waveforms to average for a math waveform.

**Group** Math

**Syntax** MATH<x>:NUMavg <NR1>  
MATH<x>:NUMavg?

**Related Commands** [MATH<x>:DEFine](#)

**Arguments** NR1 is the number of waveforms to average for a math waveform. NR1 must be in the range 2 to 1000000.

**Examples** MATH4:NUMAVG 25 sets the number of waveforms to average for the Math 4 waveform to 25.

MATH2:NUMAVG? might return MATH2:NUMAVG 15, indicating that the number of waveforms to average for the Math 2 waveform is 15.

## MATH<x>:POSition

This command allows you to set or query the vertical position of the specified math waveform. The math waveform is specified by x, which can be 1 through 8. This is equivalent to specifying a math waveform in the Waveform section of the Vertical Setup dialog box and then entering a value for Position in the Setup section.

**Group** Math

**Syntax** MATH<x>:POSITION <NR3>  
MATH<x>:POSITION?

**Related Commands** [MATH<x>:SCALE](#)

**Arguments** NR3 is the desired position in divisions from the center graticule. The range is  $\pm 1000$  divisions.

**Examples** MATH2:POSITION 1.3E+00 positions the Math 2 waveform 1.3 divisions above the center of the display.

MATH1:POSITION? might return MATH1:POSITION -1.5000000000 as the position setting for the Math 1 waveform, indicating that the waveform is positioned 1.5 divisions below the center of the display.

## MATH<x>:SCALE

This command allows you to set or query the vertical scale of the math waveform specified by x, which can be 1 through 8. This is equivalent to specifying a math waveform in the Waveform section of the Vertical Setup dialog box and then entering a value for Scale in the Setup section.

Each waveform has its own vertical scale parameter. For a signal with constant amplitude, increasing the scale causes the waveform to be displayed smaller. Decreasing the scale causes the waveform to be displayed larger. For math waveforms, this setting controls the display only, graphically scaling these waveforms and having no effect on the acquisition hardware.

**Group** Math

**Syntax** MATH<x>:SCALE <NR3>  
MATH<x>:SCALE?

**Related Commands** [MATH<x>:POSITION](#)

**Arguments** NR3 is the scale in vertical units per division. The range is 1.0E-30 to 1.0E+30.

**Examples** MATH4:SCALE 100E-03 sets the Math 4 waveform scale to 100 mV per division.

MATH1:SCALE? might return MATH1:SCALE 1.0000000000 as the scale setting for the Math 1 waveform, indicating that the scale setting of the Math 1 waveform is 1 volt per division.

## MATH<x>:UNIts? (Query Only)

This query only command returns the math units for the specified math waveform. The x argument can be 1 through 8, specifying math waveforms M1 through M8. The instrument will attempt to determine the appropriate units for the result of the math expression that generates the math waveform. The default is UNDEFINED unless one of the following rules can be applied:

- If the resultant waveform is unitless, a "U" is displayed for the unit label.
- Plus (+), Minus (-), Average, Envelope, and Filter functions use the unit label of the source operands for the vertical axis readout and the horizontal axis unit label of the source operands for the horizontal axis readout. If the units for the sources are not the same, then the result is unitless.
- An integration function concatenates the vertical axis unit label and the horizontal axis label of its source operand for the vertical axis readout (for example, Vs) and the horizontal axis unit label of the source operands for the horizontal axis readout.
- A differential function concatenates the vertical axis unit label and the horizontal axis label of its source operand with a slash (/) inserted between labels (for example, V/s) and the horizontal axis unit label of the source operands for the horizontal axis readout.
- The multiply function (\*) concatenates the vertical axis unit labels of its source operands for the vertical axis readout (for example, VV) and the horizontal axis unit label of the source operands for the horizontal axis readout.
- The divide function concatenates the unit labels of its source operands with a slash (/) inserted between the labels (for example, V/V) and the horizontal axis unit label of the source operands for the horizontal axis readout.
- The log, ln, sqrt, and  $e^x$  function results are unitless.

**Group** Math

**Syntax** MATH<x>:UNIts?

**Examples** MATH3:UNITS? might return MATH3:UNITS "V", indicating that the units for Math 3 waveform are volts.

## MATH<x>:WFMLabel

This command sets or queries the label associated with the math waveform specified.



<b>Group</b>	Math
<b>Syntax</b>	MATH<x>:WFMLabel <QString> MATH<x>:WFMLabel?
<b>Related Commands</b>	<a href="#">CH&lt;x&gt;:WFMLabel</a> , <a href="#">REF&lt;x&gt;:WFMLabel</a>
<b>Arguments</b>	QString sets the label for the math waveform.
<b>Examples</b>	MATH1:WFMLABEL "MY MATH1 WAVEFORM" sets the label for MATH1 to be "My MATH1 waveform," which is the label displayed with the waveform when it is displayed on screen.  MATH1:WFMLABEL? might return MATH1:WFMLABEL "MY MATH1 WAVEFORM", indicating that the label for the MATH1 waveform is set to "My MATH1 waveform."

## MEASUREMENT? (Query Only)

This is a query only command that returns all measurement parameters.

<b>Group</b>	Measurement
<b>Syntax</b>	MEASUREMENT?
<b>Examples</b>	MEASUREMENT? might return the following (partial list):  MEASUREMENT:ALL:VALUE 9.9100000000E+037, 9.9100000000E+037,9.9100000000E+037, 9.9100000000E+037,9.9100000000E+037, 9.9100000000E+037,9.9100000000E+037, 9.9100000000E+037; :MEASUREMENT:LIST 1; LISTVALUE 9.9100000000E+037; ANNOTATIONS:STATE 0; MEASUREMENT:STATISTICS:ENABLE 0; WEIGHTING 32; MEASUREMENT:MEAS1:NOISE HIGH; JITTER EYECROSS; EYEWINDOW 20;MAXIMUM 9.9100000000E+037; MINIMUM 9.9100000000E+037; MEAN 9.9100000000E+037; STDDEV 9.9100000000E+037;VALUE 9.9100000000E+037; ALL 9.9100000000E+037,9.9100000000E+037, 9.9100000000E+037,9.9100000000E+037, 9.9100000000E+037; UNITS "";TYPE UNDEFINED;STATE 0; GATING:STATE 0; :MEASUREMENT:MEAS1:SOURCE1:WFM CH3, MAIN;WFMDB:STATE

```
0;SIGTYPE PULSE;:MEASUREMENT:MEAS1 SOURCE1:GATE1:POS
1.80000000000E-008; ...
```

## MEASUrement:ALL:VALue? (No Query Form)

This query only command returns all the measurement values.

**Group** Measurement

**Syntax** MEASUrement:ALL:VALue?

**Related Commands** [MEASUrement:LISTValue?](#), [MEASUrement:MEAS<x>:VALue?](#)

**Examples** MEASUREMENT:ALL:VALUE? might return MEASUREMENT:ALL:VALUE  
5.18798842266E-004, 1.08967389191E-010, 99.10000000000E+36,  
1.57041666667E-009, 6.367736800021E+008,  
99.10000000000E+36, 4.11368599818E-001", 99.10000000000E+36 as  
the listing of the eight possible measurement values.

---

**NOTE.** The value 99.10000000000E+36 in the returned list of values is the numeric equivalent to NAN (not a number); this value indicates that the measurement slot does not have a valid measurement defined for it.

---

## MEASUrement:ANNOtations:STATE

This command sets or queries whether annotations are displayed. This is equivalent to toggling the Annotations check box in the Measurement Setup dialog box. To display gates used in gated measurements, Gates must be enabled with the MEASUrement:MEAS<x>:GATing:STATE command.

**Group** Measurement

**Syntax** MEASUrement:ANNOtations:STATE { ON | OFF | NR1 }  
MEASUrement:ANNOtations:STATE?

**Related Commands** [MEASUrement:MEAS<x>:GATing:STATE](#)

- Arguments**
- ON enables the display of annotations.
  - OFF disables the display of annotations.
  - NR1 set to 0 disables the display of annotations; any other integer enables the display of annotations.

**Examples** MEASUREMENT:ANNOTATIONS:STATE ON enables the display of annotations.  
 MEASUREMENT:ANNOTATIONS:STATE? might return  
 MEASUREMENT:ANNOTATIONS:STATE 0, indicating that the display of annotations is disabled.

---

**NOTE.** A query always returns a 0 or 1.

---

## MEASUREMENT:LIST

You use this command to establish a list of defined measurements for which you want values returned. The query form of this command returns the list of the defined measurements (for example, MEAS1, MEAS2). You can use the MEASUREMENT:LISTValue? query to get the value for each of the defined measurements in the list. There is no equivalent front-panel or user-interface application control for this feature.

**Group** Measurement

**Syntax** MEASUREMENT:LIST {<NR1>[, <NR1>]}  
 MEASUREMENT:LIST?

**Related Commands** [MEASUREMENT:LISTValue?](#)

**Arguments** NR1 is one of the defined measurement slots. You can create a list of up to 8 measurements (1 through 8) that correspond to Meas 1 through Meas 8.

**Examples** MEASUREMENT:LIST 1,4,5 establishes a list of 3 defined measurements (Meas 1, Meas 4, and Meas 5) for which values can be returned.  
 MEASUREMENT:LIST? might return MEASUREMENT:LIST 1,2, indicating that the list consists of two measurements.

## MEASUREMENT:LISTVALUE? (Query Only)

This query only command returns the values of the measurements in the list created with the MEASUREMENT:LIST command.

**Group** Measurement

**Syntax** MEASUREMENT:LISTVALUE?

**Related Commands** [MEASUREMENT:LIST](#)

**Examples** MEASUREMENT:LISTVALUE? might return MEASUREMENT:LISTVALUE 1.52587894784E-004, 2.49499996255E-010, which are the measurement values for the measurement slots defined as part of the list.

## MEASUREMENT:MEAS<x>:ALL? (Query Only)

This query only command returns all the statistics for the measurement specified by x, which can be 1 through 8. Before you send this command, send the [MEASUREMENT:STATISTICS:ENABLE ON](#) command to ensure accurate measurement statistics. The following measurement values are returned (in the order listed below):

- Maximum
- Minimum
- Mean
- Standard deviation
- Measurement value

**Group** Measurement

**Syntax** MEASUREMENT:MEAS<x>:ALL?

**Related Commands** [MEASUREMENT:MEAS<x>:MAXIMUM?](#), [MEASUREMENT:MEAS<x>:MINIMUM?](#), [MEASUREMENT:MEAS<x>:MEAN?](#), [MEASUREMENT:MEAS<x>:STDDEV?](#), [MEASUREMENT:MEAS<x>:VALUE?](#), [MEASUREMENT:STATISTICS:ENABLE](#)

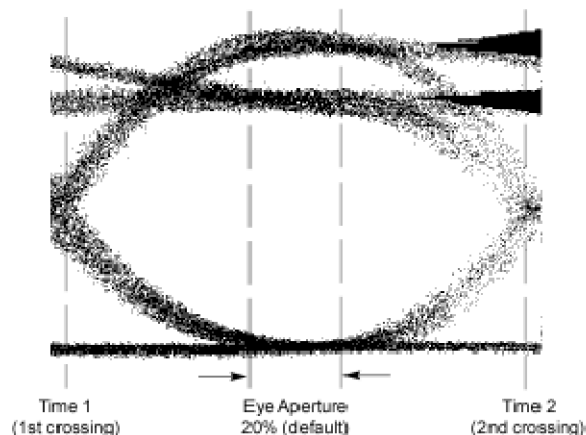
**Returns** NR3

**Examples** MEASUREMENT:MEAS1:ALL? might return MEASUREMENT:MEAS1:ALL  
 2.15759283225E-002,2.44140631654E-003,  
 3.23194688283E-003,2.67703731965E-003, 3.20434579046E-003.

## MEASUREMENT:MEAS<x>:EYEWINDOW

This command sets or queries the eye aperture value. This value is the percentage of the interval between two eye crossings (centered on the middle of the eye region) within which eye measurements are taken. (See figure below.) The interval from the first crossing to the second crossing is equivalent to 100%. Sending this command is equivalent to setting a percentage for the Aperture control in the Eye section of the Region tab in the Measurement Setup dialog box. The Eye section is displayed only if you have an eye-pattern or optical measurement defined.

Eye aperture is used for eye measurements, such as Extinction Ratio, Percent, dB, Qfactor, Eye Height and Crossing Percent.



**Group** Measurement

**Syntax** MEASUREMENT:MEAS<x>:EYEWINDOW <NR2>  
 MEASUREMENT:MEAS<x>:EYEWINDOW?

**Arguments** NR2 is the percent of interval between two eye crossings centered on the middle of the region. The range is 0 to 100. The default is 20.

**Examples** MEASUREMENT:MEAS1:EYEWINDOW 25 sets the eye aperture (window) to 25% for Measurement 1.

MEASUREMENT:MEAS1:EYEWINDOW? might return  
 MEASUREMENT:MEAS1:EYEWINDOW 10, indicating that the eye aperture is set to 10%. Measurement 1 will be taken within this region.

## MEASUREMENT:MEAS<x>:GATING:STATE

This command sets or queries the gating state for the measurement slot specified by x, which can be 1 through 8. Sending this command is equivalent to toggling the On check box in the Gates section of the Region portion of the Measurement Setup dialog box.

A gate defines a segment of waveform data over which the measurement is performed. You can define the specified measurement using the MEASUREMENT:MEAS<x>:TYPE command. You can use the MEASUREMENT:ANNOTATIONS:STATE command to display the gating in the graticule area.

**Group** Measurement

**Syntax** MEASUREMENT:MEAS<x>:GATING:STATE { ON | OFF | NR1 }  
 MEASUREMENT:MEAS<x>:GATING:STATE?

**Related Commands** [MEASUREMENT:MEAS<x>:TYPE](#), [MEASUREMENT:ANNOTATIONS:STATE](#), [MEASUREMENT:MEAS<x>:SOURCE<x>:GATE<x>:POS](#), [MEASUREMENT:MEAS<x>:SOURCE<x>:GATE<x>:PCTPOS](#)

- Arguments**
- ON enables the gating for the specified measurement.
  - OFF disables the gating for the specified measurement.
  - NR1 set to 0 disables the gating for the specified measurement; any other value enables the gating for the specified measurement.

**Examples** MEASUREMENT:MEAS2:GATING:STATE ON enables the gating for Measurement 2.

MEASUREMENT:MEAS1:GATING:STATE? might return  
 MEASUREMENT:MEAS1:GATING:STATE 1, indicating that gating is enabled for Measurement 1.

---

**NOTE.** A query always returns a 0 or 1.

---

## MEASUREMENT:MEAS<x>:JITTER

This command sets or queries whether the jitter measurement should be made at the eye crossing of an eye pattern or at the mid reference level. Sending this command is equivalent to selecting Eye Cross or Mid Ref for the Jitter At field in the Eye section of the Region tab in the Measurement Setup dialog box. The Eye section is displayed only if you have an eye-pattern or optical measurement defined.

<b>Group</b>	Measurement
<b>Syntax</b>	MEASUREMENT:MEAS<x>:JITTER {EYECross   MIDref} MEASUREMENT:MEAS<x>:JITTER?
<b>Arguments</b>	<ul style="list-style-type: none"> <li>■ EYECross causes the measurement for jitter to be taken at the eye crossing.</li> <li>■ MIDref causes the measurement for jitter to be taken at the mid reference level.</li> </ul>
<b>Examples</b>	<p>MEASUREMENT:MEAS1:JITTER EYECROSS causes the jitter measurement to be made at the eye crossing.</p> <p>MEASUREMENT:MEAS2:JITTER? might return MEASUREMENT:MEAS2:JITTER MIDREF, indicating that Measurement 2 will measure jitter at the mid reference level.</p>

## MEASUREMENT:MEAS<x>:MAXIMUM? (Query Only)

This query only command returns the maximum value found for the measurement slot specified by x, which can be 1 through 8, since the last statistical reset.

<b>Group</b>	Measurement
<b>Syntax</b>	MEASUREMENT:MEAS<x>:MAXIMUM?
<b>Related Commands</b>	<a href="#">MEASUREMENT:MEAS&lt;x&gt;:ALL?</a>
<b>Returns</b>	NR3

**Examples** MEASUREMENT:MEAS3:MAXIMUM? might return  
MEASUREMENT:MEAS3:MAXIMUM 4.27246105395E-003, indicating that the  
maximum value for Meas 3 is 4.272 mV.

## MEASUREMENT:MEAS<x>:MEAN? (Query Only)

This query only command returns the mean value accumulated for the measurement slot specified by x, which can be 1 through 8, since the last statistical reset.

**Group** Measurement

**Syntax** MEASUREMENT:MEAS<x>:MEAN?

**Related Commands** [MEASUREMENT:MEAS<x>:ALL?](#)

**Returns** NR3

**Examples** MEASUREMENT:MEAS1:MEAN? might return MEASUREMENT:MEAS1:MEAN  
2.85516357276E-003, indicating that the mean value for Meas 1 is 2.855 mV.

## MEASUREMENT:MEAS<x>:MINIMUM? (Query Only)

This query only command returns the minimum value found for the measurement slot specified by x, which can be 1 through 8, since the last statistical reset.

**Group** Measurement

**Syntax** MEASUREMENT:MEAS<x>:MINIMUM?

**Related Commands** [MEASUREMENT:MEAS<x>:ALL?](#)

**Returns** NR3

**Examples** MEASUREMENT:MEAS1:MINIMUM? might return  
MEASUREMENT:MEAS1:MINIMUM 1.92260747428E-003, indicating that the  
minimum value for Meas 1 is 1.923 mV.



## MEASUREMENT:MEAS<x>:NOISE

This command sets or queries whether the noise measurement is made on the high or low level of the waveform. Sending this command is equivalent to selecting High or Low for the Noise At field in the Eye section of the Region tab in the Measurement Setup dialog box. The Eye section is displayed only if you have an eye-pattern or optical measurement defined.

<b>Group</b>	Measurement
<b>Syntax</b>	MEASUREMENT:MEAS<x>:NOISE {HIGH   LOW} MEASUREMENT:MEAS<x>:NOISE?
<b>Arguments</b>	<ul style="list-style-type: none"> <li>■ HIGH causes the measurement for noise to be taken at the high level of the waveform.</li> <li>■ LOW causes the measurement for noise to be taken at the low level of the waveform.</li> </ul>
<b>Examples</b>	<p>MEASUREMENT:MEAS1:NOISE HIGH causes the noise measurement to be made at the high level of the waveform for Measurement 1.</p> <p>MEASUREMENT:MEAS2:NOISE? might return MEASUREMENT:MEAS2:NOISE LOW, indicating that Measurement 2 will measure noise at the low level of the waveform.</p>

## MEASUREMENT:MEAS<x>:REFLEVEL<x>? (Query Only)

This is a query only command that returns all reference level <x> settings for measurement slot <x> (slot one through slot eight).

<b>Group</b>	Measurement
<b>Syntax</b>	MEASUREMENT:MEAS<x>:REFLEVEL<x>?
<b>Examples</b>	<p>MEASUREMENT:MEAS1:REFLEVEL? might return MEASUREMENT:MEAS1:REFLEVEL1:METHOd RELATIVE; ABSOLUTE:HIGH 0.0000000000E+000; LOW 0.0000000000E+000;MID 0.0000000000E+000.</p>

## MEASUREMENT:MEAS<x>:REFLEVEL<x>:ABSOLUTE:HIGH

This command sets or queries the high reference level in absolute user units for the specified measurement. The measurement slot is specified by x, which can be 1 through 8. The reference level source is specified by x, which can only be 1 for this measurement (or defaults to 1 if not specified).

This absolute value is used as the high reference level when MEASUREMENT:MEAS<x>:REFLEVEL<x>:METHOD is set to ABSOLUTE. Sending this command is equivalent to selecting the RefLevel tab in the Measurement Setup dialog box, selecting Absolute in the Reference Level Calc Method section, and then entering an absolute value for Hi in the Reference section.

**Group** Measurement

**Syntax** MEASUREMENT:MEAS<x>:REFLEVEL<x>:ABSOLUTE:HIGH <NR3>  
MEASUREMENT:MEAS<x>:REFLEVEL<x>:ABSOLUTE:HIGH?

**Related Commands** [MEASUREMENT:MEAS<x>:REFLEVEL<x>:METHOD](#)

**Arguments** NR3 is the high reference level in absolute user units. The default is 0 V.

**Examples** MEASUREMENT:MEAS3:REFLEVEL1:ABSOLUTE:HIGH 2.0E-1 sets the high reference level for Measurement 3 to 200 mV.

MEASUREMENT:MEAS2:REFLEVEL1:ABSOLUTE:HIGH? might return  
MEASUREMENT:MEAS2:REFLEVEL1:ABSOLUTE:HIGH 9.0000000000E-2,  
indicating that the high reference level for Measurement 2 to is set to 90 mV.

## MEASUREMENT:MEAS<x>:REFLEVEL<x>:ABSOLUTE:LOW

This command sets or queries the low reference level in absolute user units for the specified measurement. The measurement slot is specified by x, which can be 1 through 8. The reference level source is specified x, which can only be 1 for this measurement (or defaults to 1 if not specified).

This absolute value is used as the low reference level when MEASUREMENT:MEAS<x>:REFLEVEL<x>:METHOD is set to ABSOLUTE. Sending this command is equivalent to selecting Absolute in the Reference Level Calc Method section of the RefLevel dialog and then entering an absolute value for Low in the Reference section of the Measurement Setup dialog box.

**Group** Measurement

**Syntax** MEASUREMENT:MEAS<x>:REFLEVEL<x>:ABSOLUTE:LOW <NR3>  
MEASUREMENT:MEAS<x>:REFLEVEL<x>:ABSOLUTE:LOW?

**Related Commands** [MEASUREMENT:MEAS<x>:REFLEVEL<x>:METHOD](#)

**Arguments** NR3 is the low reference level in absolute user units. The default is 0 V.

**Examples** MEASUREMENT:MEAS3:REFLEVEL1:ABSOLUTE:LOW 2.0E-2 sets the low reference level for Measurement 3 to 20 mV.

MEASUREMENT:MEAS2:REFLEVEL1:ABSOLUTE:LOW? might return  
MEASUREMENT:MEAS2:REFLEVEL1:ABSOLUTE:LOW 3.0000000000E-003,  
indicating that the low reference level for Measurement 2 to is set to 3 mV.

## MEASUREMENT:MEAS<x>:REFLEVEL<x>:ABSOLUTE:MID

This command sets or queries the mid reference level in absolute user units for the specified measurement. The measurement slot is specified by x, which can be 1 through 8. The reference level is specified by x (which corresponds to the Source1 or Source2 waveform) and can be 1 or 2. (You can set two different mid reference levels when taking measurements, such as phase or delay, between two source waveforms.)

This absolute value is used as the mid reference level when MEASUREMENT:MEAS<x>:REFLEVEL<x>:METHOD is set to ABSOLUTE. Sending this command is equivalent to selecting ABSOLUTE in the Reference Level Calc Method section of the RefLevel dialog and then entering an absolute value for Mid in the Reference section of the Measurement Setup dialog box.

**Group** Measurement

**Syntax** MEASUREMENT:MEAS<x>:REFLEVEL<x>:ABSOLUTE:MID <NR3>  
MEASUREMENT:MEAS<x>:REFLEVEL<x>:ABSOLUTE:MID?

**Related Commands** [MEASUREMENT:MEAS<x>:REFLEVEL<x>:METHOD](#)

**Arguments** NR3 is the mid reference level in absolute user units. The default is 0 V.

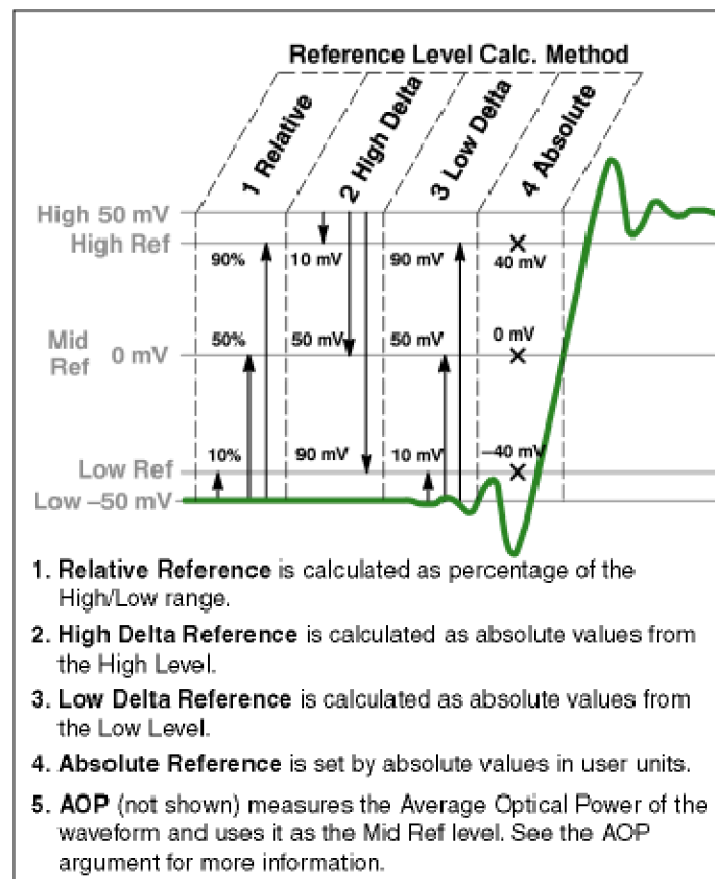
**Examples** MEASUREMENT:MEAS3:REFLEVEL2:ABSOLUTE:MID 4.0E-2 sets the mid reference level for Measurement 3 on Source 2 to 40 mV.

MEASUREMENT:MEAS2:REFLEVEL1:ABSOLUTE:MID? might return  
 MEASUREMENT:MEAS2:REFLEVEL1:ABSOLUTE:MID 5.000000000E-2,  
 indicating that the mid reference level for Measurement 2 on Source 1 is set to 50 mV.

## MEASUREMENT:MEAS<x>:REFLEVEL<x>:METHOD

This command sets or queries the method the instrument uses to calculate the reference levels for a specified measurement taken on a specified source waveform. The measurement slot is specified by x, which can be 1 through 8. The reference level is specified by x (which corresponds to the Source1 or Source2 waveform) and can be 1 or 2. Sending this command is equivalent to selecting the option button for the type of calculation in the Reference Level Calc Method section of the RefLevel dialog in the Measurement Setup dialog box.

The following diagram shows how RELative, ABSolute, HIDelta, and LODelta methods calculate the High and Low reference levels.



Group Measurement

**Syntax** MEASUREMENT:MEAS<x>:REFLEVEL<x>:METHOD {RELATIVE | ABSOLUTE | HIDEALTA | LODALTA | AOPT}  
MEASUREMENT:MEAS<x>:REFLEVEL<x>:METHOD?

**Related Commands** MEASUREMENT:MEAS<x>:REFLEVEL<x>:ABSOLUTE:HIGH, MEASUREMENT:MEAS<x>:REFLEVEL<x>:ABSOLUTE:LOW, MEASUREMENT:MEAS<x>:REFLEVEL<x>:ABSOLUTE:MID, MEASUREMENT:MEAS<x>:REFLEVEL<x>:RELATIVE:HIGH, MEASUREMENT:MEAS<x>:REFLEVEL<x>:RELATIVE:LOW, MEASUREMENT:MEAS<x>:REFLEVEL<x>:RELATIVE:MID

- Arguments**
- **RELATIVE** calculates the reference levels as a percentage of the High/Low amplitude (High amplitude minus the Low amplitude). The default values are 90% for the high reference level, 10% for the low reference level, and 50% for the mid reference levels. You can set other percentages using the MEASUREMENT:MEAS<x>:REFLEVEL:RELATIVE commands.
  - **ABSOLUTE** uses reference levels set explicitly in absolute user units with the MEASUREMENT:MEAS<x>:REFLEVEL:ABSOLUTE commands (see related commands above). This method is useful when precise values are required (for example, when you are designing to published interface specifications such as RS-232-C). The default values are 0 V for the high reference level, the low reference level, and the mid reference levels.
  - **HIDEALTA** sets the reference levels as absolute amplitude delta values from the High level.
  - **LODALTA** sets the reference levels as absolute amplitude delta values from the Low level.
  - **AOPT** sets the reference level to best measure Optical Modulation Amplitude (OMA) on Pulse waveforms, and is the default selection when taking OMA measurements on Pulse or NRZ waveforms. OMA measurements are described in the User Online Help, which is accessible from the instrument Help menu.

**Examples** MEASUREMENT:MEAS1:REFLEVEL1:METHOD RELATIVE sets the method of calculating the reference levels to relative for Measurement 1; the default values are 90% for the high reference level, 50% for the mid reference level, and 10% for the low reference level.

MEASUREMENT:MEAS8:REFLEVEL1:METHOD? might return  
MEASUREMENT:MEAS8:REFLEVEL1:METHOD ABSOLUTE, indicating the reference levels used are set to absolute values in user units.

## MEASUrement:MEAS<x>:REFLevel<x>:RELative:HIGH

This command sets or queries the percent of the High/Low range that the instrument uses to calculate the high reference level for the specified measurement, where 100% is equal to the High/Low range. The measurement slot is specified by x, which can be 1 through 8. The reference level source is specified x, which can only be 1 for this measurement (or defaults to 1 if not specified).

This percentage is used to calculate the high reference level when MEASUrement:MEAS<x>:REFLevel<x>:METHod is set to RELative. Sending this command is equivalent to selecting Relative in the Reference Level Calc Method section of the RefLevel dialog and then entering a percentage for Hi in the Reference section of the Measurement Setup dialog box.

**Group** Measurement

**Syntax** MEASUrement:MEAS<x>:REFLevel<x>:RELative:HIGH <NR2>  
MEASUrement:MEAS<x>:REFLevel<x>:RELative:HIGH?

**Related Commands** [MEASUrement:MEAS<x>:REFLevel<x>:METHod](#)

**Arguments** NR2 can be from 0 to 100 (percent) and is the high reference level. The default is 90 (percent).

**Examples** MEASUREMENT:MEAS3:REFLEVEL1:RELATIVE:HIGH 95 sets the high reference level for Measurement 3 to 95% of the High/Low range.

MEASUREMENT:MEAS2:REFLEVEL1:RELATIVE:HIGH? might return  
MEASUREMENT:MEAS2:REFLEVEL1:PERCENT:HIGH 9.0000000000E+001,  
indicating that the high reference level for Measurement 2 to is set to 90% (the default value) of the High/Low range.

## MEASUrement:MEAS<x>:REFLevel<x>:RELative:LOW

This command sets or queries the percent of the High/Low range that the instrument uses to calculate the low reference level for the specified measurement, where 100% is equal to the High/Low range. The measurement slot is specified by x, which can be 1 through 8. The reference level source is specified x, which can only be 1 for this measurement (or defaults to 1 if not specified).

This percentage is used to calculate the low reference level when MEASUrement:MEAS<x>:REFLevel<x>:METHod is set to RELative. Sending this command is equivalent to selecting Relative in the Reference Level Calc

Method section of the RefLevel dialog and then entering a percentage for Low in the Reference section of the Measurement Setup dialog box.

**Group** Measurement

**Syntax** MEASUREMENT:MEAS<x>:REFLEVEL<x>:RELATIVE:LOW <NR2>  
MEASUREMENT:MEAS<x>:REFLEVEL<x>:RELATIVE:LOW?

**Related Commands** [MEASUREMENT:MEAS<x>:REFLEVEL<x>:METHOD](#)

**Arguments** NR2 can be from 0 to 100 (percent) and is the low reference level. The default is 10 (percent).

**Examples** MEASUREMENT:MEAS3:REFLEVEL1:RELATIVE:LOW 20 sets the low reference level for Measurement 3 to 20% of the High/Low range.

MEASUREMENT:MEAS2:REFLEVEL1:RELATIVE:LOW? might return  
MEASUREMENT:MEAS2:REFLEVEL1:RELATIVE:LOW 10, indicating that the low reference level for Measurement 2 is set to 10% (the default value) of the High/Low range.

## MEASUREMENT:MEAS<x>:REFLEVEL<x>:RELATIVE:MID

This command sets or queries the percent of the High/Low range that the instrument uses to calculate the mid reference level for the specified measurement, where 100% is equal to the High/Low range. The measurement slot is specified by x, which can be 1 through 8. The reference level is specified by x (which corresponds to the Source1 or Source2 waveform) and can be 1 or 2. (You can set two different mid reference levels when taking measurements, such as phase or delay, between two source waveforms.)

This percentage is used to calculate the mid reference level when MEASUREMENT:MEAS<x>:REFLEVEL<x>:METHOD is set to Relative. Sending this command is equivalent to selecting Relative in the Reference Level Calc Method section of the RefLevel dialog and then entering a percentage for Mid in the Reference section of the Measurement Setup dialog box.

**Group** Measurement

**Syntax** MEASUREMENT:MEAS<x>:REFLEVEL<x>:RELATIVE:MID <NR2>  
MEASUREMENT:MEAS<x>:REFLEVEL<x>:RELATIVE:MID?

**Related Commands**    [MEASUREMENT:MEAS<x>:REFLEVEL<x>:METHOD](#)

**Arguments**    NR2 can be from 0 to 100 (percent) and is the mid reference level. The default is 50 (percent).

**Examples**    MEASUREMENT:MEAS3:REFLEVEL2:RELATIVE:MID 40 sets the mid reference level for Measurement 3 on Source 2 to 40% of the High/Low range.

MEASUREMENT:MEAS2:REFLEVEL1:RELATIVE:MID? might return  
MEASUREMENT:MEAS2:REFLEVEL1:RELATIVE:MID 5.0000000000E+001,  
indicating that the mid reference level for Measurement 2 on Source 1 is set to 50% (the default value) of the High/Low range.

## MEASUREMENT:MEAS<x>:SETDEFAULT (No Query Form)

This command sets the Region, Reflevel, and HiLow parameters for measurement <x> to the factory default settings. This command does not affect measurement source or type settings.

**Group**    Measurement

**Syntax**    MEASUREMENT:MEAS<x>:SETDEFAULT

**Examples**    MEASUREMENT:MEAS3:SETDEFAULT sets the Region, Reflevel, and HiLow parameters for Measurement 3 to their factory default settings.

## MEASUREMENT:MEAS<x>:SOURCE<x>:EDGE? (Query Only)

This query only command returns edge parameters (slope and direction) for the specified measurement slot using the specified source. The measurement slot is specified by x, which can be 1 through 8. The Source is specified by x, which can be 1 or 2.

---

**NOTE.** The SOURCE<x> argument specifies the source, which can be source 1 or source 2. To set the actual waveform (channel, reference, or math) that is assigned to source, you must use the MEASUREMENT:MEAS<x>:SOURCE<x>WFM command. Two sources are provided because some measurements (gain, phase) require two sources.

---

**Group**    Measurement



**Syntax**    MEASUREMENT:MEAS<x>:SOURCE<x>:EDGE?

**Examples**    MEASUREMENT:MEAS4:SOURCE1:EDGE? might return  
 MEASUREMENT:MEAS4:SOURCE1:EDGE:SLOPE FALL;DIRECTION FORWARD.

## MEASUREMENT:MEAS<x>:SOURCE<x>:EDGE:DIRection

This command sets or queries the direction that the instrument uses to look for the rising or falling edge of the specified waveform when taking the specified measurement. This command affects only Delay, Ncross and Pcross measurements. The measurement slot is specified by x, which can be 1 through 8; the source waveform is specified by x, which can be 1 or 2.

Sending this command is equivalent to selecting Setup > Measurement tab > Region tab > Edges > direction button.

---

**NOTE.** The SOURCE<x> argument specifies the source, which can be source 1 or source 2. To set the actual waveform (channel, reference, or math) that is assigned to source, you must use the MEASUREMENT:MEAS<x>:SOURCE<x>WFM command. Two sources are provided because some measurements (gain, phase) require two sources.

---

**Group**    Measurement

**Syntax**    MEASUREMENT:MEAS<x>:SOURCE<x>:EDGE:DIRection { FORWARD |  
 BACKWARD }  
 MEASUREMENT:MEAS<x>:SOURCE<x>:EDGE:DIRection?

**Related Commands**    [MEASUREMENT:MEAS<x>:GATING:STATE](#), [MEASUREMENT:MEAS<x>:SOURCE<x>:EDGE:SLOPe](#)

- Arguments**
- FORWARD causes the instrument to search forward from the beginning of the waveform (or gated region) and find the first edge with the slope set by the MEASUREMENT:MEAS<x>:SOURCE<x>:EDGE:SLOPe command (delay measurements) or for the edge implicitly determined by the defined measurement.
  - BACKWARD causes the instrument to search backward from the end of the waveform (or gated region) and find the last edge with the slope set by the MEASUREMENT:MEAS<x>:SOURCE<x>:EDGE:SLOPe command (delay measurements) or for the edge implicitly determined by the defined measurement.

**Examples** MEASUREMENT:MEAS2:SOURCE1:EDGE:DIRECTION FORWARD causes the instrument to search forward from the beginning of the Source 1 waveform (or the beginning of the gated region, if gates are enabled) for the first specified edge (rising or falling).

MEASUREMENT:MEAS8:SOURCE1:EDGE:DIRECTION? might return MEASUREMENT:MEAS8:SOURCE1:EDGE:DIRECTION FORWARD, indicating that the instrument will search forward from the beginning of the Source 1 waveform (or the beginning of the gated region, if gates are enabled) for the specified edge (rising or falling) when taking Measurement 8.

## MEASUREMENT:MEAS<x>:SOURCE<x>:EDGE:SLOPE

This command sets or queries the slope (or polarity) of the edges used in delay time measurements. The measurement slot is specified by x, which can be 1 through 8; the source is specified by x, which can be 1 (source waveform) or 2 (destination waveform). For edge measurements other than delay, the slope is determined implicitly (for example, rise time measurements use positive slopes). If gates are enabled (see the [MEASUREMENT:MEAS<x>:GATING:STATE](#) command), the instrument finds the edge within the measurement zone; otherwise, the instrument finds the edge within the entire waveform record.

Sending this command is equivalent to selecting an option button ( $\pm$ , +, or -) for Slope in the Edges section of the Region dialog in the Measurement Setup dialog box. (A delay measurement must be selected before these buttons are active.)

Use the MEASUREMENT:MEAS<x>:SOURCE:EDGE:DIRection command to set the direction that the instrument uses to look for the rising or falling edge of the waveforms.

---

**NOTE.** The SOURCE<x> argument specifies the source, which can be source 1 or source 2. To set the actual waveform (channel, reference, or math) that is assigned to source, you must use the MEASUREMENT:MEAS<x>:SOURCE<x>WFM command. Two sources are provided because some measurements (gain, phase) require two sources.

---

**Group** Measurement

**Syntax** MEASUREMENT:MEAS<x>:SOURCE<x>:EDGE:SLOPE {RISE | FALL | EITHER}  
 MEASUREMENT:MEAS<x>:SOURCE<x>:EDGE:SLOPE?

- Arguments**
- RISE causes the instrument to take the specified measurement on a rising (positive-going) edge of the specified source waveform.
  - FALL causes the instrument to take the specified measurement on a falling (negative-going) edge of the specified source waveform.
  - EITHER causes the instrument to take the specified measurement on either a rising (positive-going) or falling (negative-going) edge of the specified source waveform.

**Examples**

MEASUREMENT:MEAS4:SOURCE1:EDGE:SLOPE RISE causes the instrument to take the delay for Measurement 4 from a rising edge of the Source 1 waveform.

MEASUREMENT:MEAS4:SOURCE2:EDGE:SLOPE? might return  
 MEASUREMENT:MEAS4:SOURCE2:EDGE:SLOPE FALL, indicating that the delay for Measurement 4 is taken to a falling edge of the Source 2 (destination) waveform.

## MEASUREMENT:MEAS<x>:SOURCE<x>:GATE<x>? (Query Only)

This query-only command returns all gate settings of the specified gate for the specified measurement on the specified source waveform. The measurement slot is specified by x, which can be 1 through 8; the source is specified by x, which can be 1 or 2; and the gate is specified by x, which can be 1 or 2.

---

**NOTE.** The SOURCE<x> argument specifies the source, which can be source 1 or source 2. To set the actual waveform (channel, reference, or math) that is assigned to source, you must use the MEASUREMENT:MEAS<x>:SOURCE<x>WFM command. Two sources are provided because some measurements (gain, phase) require two sources.

---

**Group** Measurement

**Syntax** MEASUREMENT:MEAS<x>:SOURCE<x>:GATE<x>?

**Examples**

MEASUREMENT:MEAS1:SOURCE1:GATE1? might return  
 MEASUREMENT:MEAS1:SOURCE1:GATE1 POS 20.25000000000E-9;PCTPOS 25.0000000000.

## MEASUREMENT:MEAS<x>:SOURCE<x>:GATE<x>:PCTPOS

This command sets or queries the position (in percentage of the waveform) of the source gates for the measurement. The measurement, source, and gate are defined as follows:

- MEAS<x> where x can be 1 through 8 (a maximum of 8 measurement slots)
- SOURCE<x> where x can be 1 or 2 (a maximum of 2 source waveforms)
- GATE<x> where x can be 1 or 2 (one pair of gates per source waveform)

Gates are generic, nondirectional markers that segment the area of waveform data over which measurements are performed. There is no distinction in gates of stop or start, so they can cross.

You use this command to set the position of a gate as a percentage of the waveform. This is equivalent to selecting the Region dialog in the Measurement Setup dialog box and entering the position as a percentage of the waveform in the G1 or G2 control in the Gates section.

Note that a user cannot directly set the position of a gate in the Measurement Setup dialog box using waveform units. Instead, the user sets the percentage of the waveform at which the gate is positioned, and then the position in waveform units is displayed in the readout boxes adjacent to the G1 and G2 controls. However, you can directly set the gate position in waveform units with the GPIB command, MEASUREMENT:MEAS<x>:SOURCE<x>:GATE<x>:POS.

---

**NOTE.** The SOURCE<x> argument specifies the source, which can be source 1 or source 2. To set the actual waveform (channel, reference, or math) that is assigned to source, you must use the MEASUREMENT:MEAS<x>:SOURCE<x>:WFM command. Two sources are provided because some measurements (gain, phase) require two sources.

---

<b>Group</b>	Measurement
<b>Syntax</b>	MEASUREMENT:MEAS<x>:SOURCE<x>:GATE<x>:PCTPOS <NR2> MEASUREMENT:MEAS<x>:SOURCE<x>:GATE<x>:PCTPOS?
<b>Related Commands</b>	<a href="#">MEASUREMENT:MEAS&lt;x&gt;:GATING:STATE</a>
<b>Arguments</b>	NR2 is the position of the specified gate in percentage.
<b>Examples</b>	MEASUREMENT:MEAS4:SOURCE1:GATE1:PCTPOS 25 sets Gate 1 to 25% on the Source 1 waveform used for Measurement 4.

MEASUREMENT:MEAS2:SOURCE1:GATE2:PCTPOS? might return  
 MEASUREMENT:MEAS2:SOURCE1:GATE2:PCTPOS 8.0000000000E+001,  
 indicating that Gate 2 is set at 80% on the Source 1 waveform used for  
 Measurement 2.

## MEASUREMENT:MEAS<x>:SOURCE<x>:GATE<x>:POS

This command sets or queries the position (in waveform units) of the source gates for the measurement. The measurement, source, and gate are defined as follows:

- MEAS<x> where x can be 1 through 8 (a maximum of 8 measurement slots)
- SOURCE<x> where x can be 1 or 2 (a maximum of 2 source waveforms)
- GATE<x> where x can be 1 or 2 (one pair of gates per source waveform)

Gates are generic, nondirectional markers that segment the area of waveform data over which measurements are performed. There is no distinction in gates of stop or start, so they can cross.

Although you can use this command to set the position of a gate in waveform units, a user cannot directly set the position of a gate in the Measurement Setup dialog box using waveform units. Instead, the user selects the Region dialog, enters the position as a percentage of the waveform, and then reads the position in waveform units in the readouts adjacent to the percentage controls for G1 and G2. This is equivalent to setting the position with the MEASUREMENT:MEAS<x>:SOURCE<x>:GATE<x>:PCTPOS command.

---

**NOTE.** The SOURCE<x> argument specifies the source, which can be source 1 or source 2. To set the actual waveform (channel, reference, or math) that is assigned to source, you must use the MEASUREMENT:MEAS<x>:SOURCE<x>:WFM command. Two sources are provided because some measurements (gain, phase) require two sources.

---

**Group** Measurement

**Syntax** MEASUREMENT:MEAS<x>:SOURCE<x>:GATE<x>:POS <NR3>  
 MEASUREMENT:MEAS<x>:SOURCE<x>:GATE<x>:POS?

**Related Commands** [MEASUREMENT:MEAS<x>:SOURCE<x>:GATE<x>:PCTPos](#), [MEASUREMENT:MEAS<x>:GATING:STATE](#)

**Arguments** NR3 is the position of the specified gate in waveform units. The position is constrained to be  $\geq$ TOFP (time of first point) and  $\leq$ TOLP (time of last point).

**Examples** MEASUREMENT:MEAS4:SOURCE1:GATE1:POS 1.9E-8 sets Gate 1 to 19 ns on the Source 1 waveform used for Measurement 4 relative to the trigger.

MEASUREMENT:MEAS2:SOURCE1:GATE2:POS? might return  
 MEASUREMENT:MEAS2:SOURCE1:GATE2:POS 2.29100000000E-008,  
 indicating that Gate 2 is set at 22.91 ns on the Source 1 waveform used for Measurement 2.

## MEASUREMENT:MEAS<x>:SOURCE<x>:HILOW? (Query Only)

This query only command returns all the High/Low settings for the specified measurement taken on the specified source waveform. The measurement slot is specified by x, which can be 1 through 8; the source is specified by x, which can be 1 or 2.

---

**NOTE.** The SOURCE<x> argument specifies the source, which can be source 1 or source 2. To set the actual waveform (channel, reference, or math) that is assigned to source, you must use the MEASUREMENT:MEAS<x>:SOURCE<x> WFM command. Two sources are provided because some measurements (gain, phase) require two sources.

---

**Group** Measurement

**Syntax** MEASUREMENT:MEAS<x>:SOURCE<x>:HILOW?

**Examples** MEASUREMENT:MEAS1:SOURCE1:HIGLOW? might return  
 MEASUREMENT:MEAS1:SOURCE1:HILOW:METHOd AUTO;  
 TRACK:HIGh:ENABLE 1;VALUE 2.00000000000E-001;  
 MEASUREMENT:MEAS1:SOURCE1:HILOW:TRACK:LOW:ENABLE 1;  
 VALUE 1.00000000000E-002.

## MEASUREMENT:MEAS<x>:SOURCE<x>:HILOW:METHOd

This command sets or queries the method of calculating the high/low levels for a specified measurement on a specified source waveform. The measurement slot is specified by x, which can be 1 through 8; the source is specified by x, which can be 1 or 2. Sending this command is equivalent to specifying a measurement slot (such as Meas 3), selecting a source (such as Source1), and then checking the radio button for the desired Tracking Method in the in the Hi/Low pane of the Measurement Setup dialog box.

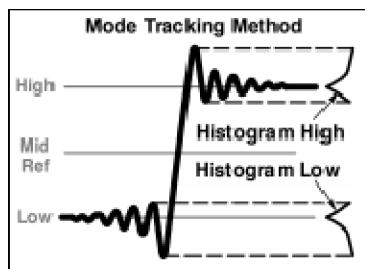
The exact value of High and Low depends on which calculation method you set.

**NOTE.** The *SOURCE*<x> argument specifies the source, which can be source 1 or source 2. To set the actual waveform (channel, reference, or math) that is assigned to source, you must use the *MEASUREMENT:MEAS*<x>:*SOURCE*<x> *WFM* command. Two sources are provided because some measurements (gain, phase) require two sources.

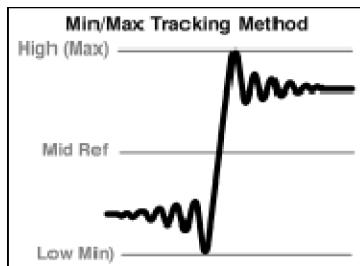
**Group** Measurement

**Syntax** `MEASUREMENT:MEAS<x>:SOURCE<x>:HILOW:METHOD {AUTO | MODE | MINMAX | MEAN}`  
`MEASUREMENT:MEAS<x>:SOURCE<x>:HILOW:METHOD?`

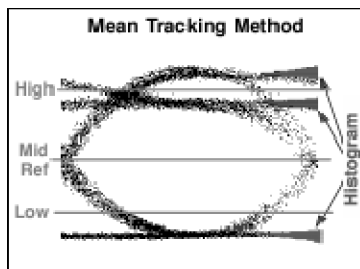
- Arguments**
- **AUTO** attempts to use the most appropriate method of calculating the high and low values for the specified signal type. If the signal type is set to Eye, the instrument defaults to the Mean method of calculating the high and low values. If the signal type is set to Pulse, the instrument defaults to the Mode method of calculating the high and low values. However, the instrument will automatically switch to the Min/Max method if the histogram used to track the high and low values does not show an obvious consistent high level or to the Mean method if the histogram shows at least two different concentrations of peak values. For example, the Mode histogram operating on a triangle wave would not find significant high and low levels, and the instrument would switch to the Min/Max mode. On a square wave, the Auto mode would use the Mode method to calculate the high and low values.
  - **MODE** attempts to find, using a histogram, the highest density of points above and below the waveform midpoint. see figure below. It attempts to ignore ringing and spikes when determining the 0% and 100% levels. This method works well when measuring square waves and pulse waveforms.



- **MINMAX** defines the 0% and 100% waveform levels as the lowest amplitude (most negative) and the highest amplitude (most positive) samples. See the figure below. This method is useful for measuring frequency, width, and period for many types of signals. However, this method is sensitive to waveform ringing and spikes and does not always accurately measure rise time, fall time, overshoot, and undershoot.



- MEAN is commonly used for eye diagrams. The mean method calculates the high value as the mean value of the region above the mid reference and the low value as the mean value of the region below the mid reference. See figure below. The mean is computed only over the eye aperture - typically the center 20% of the NRZ eye, or the center 5% of an RZ eye.



**Examples** MEASUREMENT:MEAS2:SOURCE1:HIGLOW:METHOD MINMAX selects the Min/Max method of determining the high and low values used to take measurements defined by Meas2 on the Source 1 waveform.

MEASUREMENT:MEAS1:SOURCE1:HIGLOW:METHOD? might return MEASUREMENT:MEAS1:SOURCE1:HIGLOW:METHOD MINMAX, indicating that the Min/Max method of calculating the High/Low levels is used for Meas1 on the Source 1 waveform.

## MEASUrement:MEAS<x>:SOUrce<x>:HILow:TRAcK:HIGH:ENABle

This command sets or queries whether tracking of the high value for a specified measurement on a specified source waveform is automatically done by the instrument. The measurement slot is specified by x, which can be 1 through 8; the source is specified by x, which can be 1 or 2. Sending this command is equivalent to specifying a measurement slot (such as Meas 3), selecting a source (such as Source1), and then checking the control box for Track High in the in the Hi/Low pane of the Measurement Setup dialog box. (If tracking is enabled, the instrument automatically calculates the high value using the method set with the command, MEASUrement:MEAS<x>:SOUrce<x>:HILow:METhod.)



---

**NOTE.** The *SOURCE*<x> argument specifies the source, which can be source 1 or source 2. To set the actual waveform (channel, reference, or math) that is assigned to source, you must use the *MEASUREMENT:MEAS*<x>:*SOURCE*<x> *WFM* command. Two sources are provided because some measurements (gain, phase) require two sources.

---

**Group** Measurement

**Syntax** `MEASUREMENT:MEAS<x>:SOURCE<x>:HILOW:TRACK:HIGH:ENABLE { ON | OFF | NR1 }`  
`MEASUREMENT:MEAS<x>:SOURCE<x>:HILOW:TRACK:HIGH:ENABLE?`

**Related Commands** [MEASUREMENT:MEAS<x>:SOURCE<x>:HILOW:METHOd](#), [MEASUREMENT:MEAS<x>:SOURCE<x>:HILOW:TRACK:LOW:ENABLE](#)

- Arguments**
- ON enables tracking of the high value for the specified measurement on the specified source waveform.
  - OFF disables tracking of the high value for the specified measurement on the specified source waveform.
  - NR1 set to 0 disables tracking of the high value for the specified measurement on the specified source waveform; any other value enables tracking of the high value for the specified measurement on the specified source waveform.

**Examples**

`MEASUREMENT:MEAS2:SOURCE1:HILOW:TRACK:HIGH:ENABLE ON` enables tracking of the high value used to take Measurement 2 on the Source 1 waveform.

`MEASUREMENT:MEAS3:SOURCE2:HILOW:TRACK:HIGH:ENABLE?` might return `MEASUREMENT:MEAS3:SOURCE2:HILOW:TRACK:HIGH:ENABLE 0`, indicating that the tracking of the high value for Measurement 3 on the Source 2 waveform is disabled.

---

**NOTE.** A query always returns a 0 or 1.

---

## MEASUREMENT:MEAS<x>:SOURCE<x>:HILOW:TRACK:HIGH:VALUE

This command sets or queries the high value used to calculate a specified measurement on a specified source waveform. The measurement slot is specified by x, which can be 1 through 8; the source is specified by x, which can be 1 or 2. Sending this command is equivalent to selecting a measurement slot (such as Meas 3), selecting a source (such as Source1), turning tracking High off, and then entering an absolute value in the High control box in the in the Hi/Low pane of the

Measurement Setup dialog box. (If tracking is enabled for High, the instrument automatically calculates the high value using the method set with the command, MEASUREMENT:MEAS<x>:SOURCE<x>:HILOW:METHOD.)

---

**NOTE.** The SOURCE<x> argument specifies the source, which can be source 1 or source 2. To set the actual waveform (channel, reference, or math) that is assigned to source, you must use the MEASUREMENT:MEAS<x>:SOURCE<x>:WFM command. Two sources are provided because some measurements (gain, phase) require two sources.

---

**Group** Measurement

**Syntax** MEASUREMENT:MEAS<x>:SOURCE<x>:HILOW:TRACK:HIGH:VALUE <NR3  
MEASUREMENT:MEAS<x>:SOURCE<x>:HILOW:TRACK:HIGH:VALUE?

**Related Commands** [MEASUREMENT:MEAS<x>:SOURCE<x>:HILOW:METHOD](#), [MEASUREMENT:MEAS<x>:SOURCE<x>:HILOW:TRACK:HIGH:ENABLE](#)

**Arguments** NR3 is the absolute value of the high level for the HiLow settings.

**Examples** MEASUREMENT:MEAS2:SOURCE1:HILOW:TRACK:HIGH:VALUE 5.0E-1 sets the high value used to take Measurement 2 on the Source 1 waveform to 500 mV.

MEASUREMENT:MEAS1:SOURCE1:HILOW:TRACK:HIGH:VALUE? might return MEASUREMENT:MEAS1:SOURCE1:HILOW:TRACK:HIGH VALUE 2.0000000000E-001, indicating that the high value used to take Measurement 1 on the Source 1 waveform is 200 mV.

## MEASUREMENT:MEAS<x>:SOURCE<x>:HILOW:TRACK:LOW:ENABLE

This command sets or queries whether tracking of the low value for a specified measurement on a specified source waveform is automatically done by the instrument. The measurement slot is specified by x, which can be 1 through 8; the source is specified by x, which can be 1 or 2. Sending this command is equivalent to specifying a measurement slot (such as Meas 3), selecting a source (such as Source1), and then checking the control box for Track Low in the HiLow dialog of the Measurement Setup dialog box. (If tracking is enabled, the instrument automatically calculates the low value using the method set with the command, MEASUREMENT:MEAS<x>:SOURCE<x>:HILOW:METHOD.)

---

**NOTE.** The *SOURCE*<x> argument specifies the source, which can be source 1 or source 2. To set the actual waveform (channel, reference, or math) that is assigned to source, you must use the *MEASUREMENT:MEAS*<x>:*SOURCE*<x> *WFM* command. Two sources are provided because some measurements (gain, phase) require two sources.

---

**Group** Measurement

**Syntax** `MEASUREMENT:MEAS<x>:SOURCE<x>:HILOW:TRACK:LOW:ENABLE { ON | OFF | NR1 }`  
`MEASUREMENT:MEAS<x>:SOURCE<x>:HILOW:TRACK:LOW:ENABLE?`

**Related Commands** `MEASUREMENT:MEAS<x>:SOURCE<x>:HILOW:METHOD`, `MEASUREMENT:MEAS<x>:SOURCE<x>:HILOW:TRACK:HIGHIGH:ENABLE`

- Arguments**
- ON enables tracking of the low value for the specified measurement on the specified source waveform.
  - OFF disables tracking of the low value for the specified measurement on the specified source waveform.
  - NR1 set to 0 disables tracking of the low value for the specified measurement on the specified source waveform; any other value enables tracking of the low value for the specified measurement on the specified source waveform.

**Examples**

`MEASUREMENT:MEAS2:SOURCE1:HILOW:TRACK:LOW:ENABLE ON` enables tracking of the low value used to take Measurement 2 on the Source 1 waveform.

`MEASUREMENT:MEAS3:SOURCE2:HILOW:TRACK:LOW:ENABLE?` might return `MEASUREMENT:MEAS3:SOURCE2:HILOW:TRACK:LOW:ENABLE 0`, indicating that the tracking of the low value for Measurement 3 on the Source 2 waveform is disabled.

---

**NOTE.** A query always returns a 0 or 1.

---

## MEASUREMENT:MEAS<x>:SOURCE<x>:HILOW:TRACK:LOW:VALUE

This command sets or queries the low value used to calculate a specified measurement on a specified source waveform. The measurement slot is specified by x, which can be 1 through 8; the source is specified by x, which can be 1 or 2. Sending this command is equivalent to selecting a measurement slot (such as Meas 3), selecting a source (such as Source1), setting tracking Low to off, and then entering an absolute value in the Low control box in the in the Hi/Low

pane of the Measurement Setup dialog box. (If tracking is enabled for Low, the instrument automatically calculates the low value using the method set with the command, MEASUREMENT:MEAS<x>:SOURCE<x>:HILOW:METHOD.)

---

**NOTE.** The SOURCE<x> argument specifies the source, which can be source 1 or source 2. To set the actual waveform (channel, reference, or math) that is assigned to source, you must use the MEASUREMENT:MEAS<x>:SOURCE<x>:WFM command. Two sources are provided because some measurements (gain, phase) require two sources.

---

**Group** Measurement

**Syntax** MEASUREMENT:MEAS<x>:SOURCE<x>:HILOW:TRACK:LOW:VALUE <NR3>  
MEASUREMENT:MEAS<x>:SOURCE<x>:HILOW:TRACK:LOW:VALUE?

**Related Commands** [MEASUREMENT:MEAS<x>:SOURCE<x>:HILOW:METHOD](#), [MEASUREMENT:MEAS<x>:SOURCE<x>:HILOW:TRACK:LOW:ENABLE](#)

**Arguments** NR3 is the absolute value of the low level for the HiLow settings.

**Examples** MEASUREMENT:MEAS2:SOURCE1:HILOW:TRACK:LOW:VALUE 5.0E-3 sets the low value used to take Measurement 2 on the Source 1 waveform to 5 mV.

MEASUREMENT:MEAS3:SOURCE1:HILOW:TRACK:LOW:VALUE? might return MEASUREMENT:MEAS3:SOURCE1 HILOW:TRACK:LOW:VALUE 1.0000000000E-002, indicating that the low value used to take Measurement 3 on the Source 1 waveform is 10 mV.

## MEASUREMENT:MEAS<x>:SOURCE<x>:WFM

This command sets or queries the source waveform (SOURCE1) on which the specified single-channel measurements are taken and **from which** two-channel measurements, such as delay or phase are taken. The measurement slot is specified by x, and can be 1 through 8. This command also sets or queries the destination waveform (SOURCE2) **to which** two-channel measurements are taken. If the source is not specified (either 1 or 2), the default is SOURCE1. Optionally, this command can specify the timebase for the specified source or destination waveform. If the timebase is not specified, the default is the Main timebase.

**Group** Measurement

**Syntax** MEASUREMENT:MEAS<x>:SOURCE<x>:WFM {CH<x>} | MATH<x> | REF<x>}[, {MAIN | MAG1 | MAG2}]  
MEASUREMENT:MEAS<x>:SOURCE<x>:WFM?

- Arguments**
- CH<x> selects a channel waveform as the source or destination waveform. The range for x is 1 through 8.
  - MATH<x> selects a math waveform as the source or destination waveform. The range for x is 1 through 8.
  - REF<x> selects a reference waveform as the source or destination waveform. The range for x is 1 through 8.
  - MAIN (optional) selects Main as the timebase for the specified source or destination waveform. This is the default timebase.
  - MAG1 (optional) selects MAG1 as the timebase for the specified source or destination waveform (for channels and math).
  - MAG2 (optional) selects MAG2 as the timebase for the specified source or destination waveform for channels and math).

**Examples** MEASUREMENT:MEAS2:SOURCE1:WFM MATH1 sets MATH1 as the source waveform for Measurement 2

MEASUREMENT:MEAS2:SOURCE1:WFM? might return

MEASUREMENT:MEAS2:SOURCE1:WFM CH3,MAIN, indicating that the source waveform for Measurement 2 is Channel 3 using the Main timebase.

## MEASUREMENT:MEAS<x>:SOURCE<x>:WFMDB:SIGType

This command sets or queries the type of input signal used to populate the waveform database associated with the specified measurement slot, MEAS<x>, where x can be 1 through 8, and the specified measurement source, SOURCE<x>, where x can be 1 or 2. To ensure accurate measurement results, always use this command to specify the input-signal type of the measurement source if your measurement is taken on a waveform database. Sending this command is equivalent to selecting Pulse, NRZ (Non-Return to Zero), or RZ (Return to Zero) as the Signal Type in the Source pane of the Measurement Setup dialog box.

---

**NOTE.** The SOURCE<x> argument specifies the source, which can be source 1 or source 2. To set the actual waveform (channel, reference, or math) that is assigned to source, you must use the MEASUREMENT:MEAS<x>:SOURCE<x>:WFM command. Two sources are provided because some measurements (gain, phase) require two sources.

---

<b>Group</b>	Measurement
<b>Syntax</b>	<pre>MEASUREMENT:MEAS&lt;x&gt;:SOURCE&lt;x&gt;:WFMDB:SIGType {PULSE   EYE   RZ} MEASUREMENT:MEAS&lt;x&gt;:SOURCE&lt;x&gt;:WFMDB:SIGType?</pre>
<b>Related Commands</b>	<a href="#">MEASUREMENT:MEAS&lt;x&gt;:SOURCE&lt;x&gt;:WFMDB:STATE</a>
<b>Arguments</b>	<ul style="list-style-type: none"> <li>■ PULSE signal type is used for generic signals that are not associated with synchronous communications standards.</li> <li>■ EYE signal type is for synchronous-communication signals with NRZ (non-return to zero) characteristics.</li> <li>■ RZ signal type is for synchronous communication signals with RZ (return to zero) characteristics.</li> </ul>
<b>Examples</b>	<p>MEASUREMENT:MEAS5:SOURCE1:WFMDB:SIGTYPE PULSE selects Pulse as the signal type associated with source 1 for measurement slot 5.</p> <p>MEASUREMENT:MEAS2:SOURCE1:WFMDB:SIGTYPE? might return          MEASUREMENT:MEAS2:SOURCE1 WFMDB:SIGTYPE RZ, indicating that the signal type associated with source 1 of measurement slot 2 is RZ.</p>

## MEASUREMENT:MEAS<x>:SOURCE<x>:WFMDB:STATE

This command sets or queries whether the specified measurement is taken on the specified waveform or a waveform database built from the targeted source. The measurement slot is specified by *x*, which can be 1 through 8; the source waveform is specified by *x*, which can be 1 or 2. Sending this command is equivalent to selecting a measurement and a source in the Measurement Setup dialog box, and then selecting or clearing the Use Wfm Database check box in the Source dialog of the Measurement Setup dialog box.

---

**NOTE.** The *SOURCE<x>* argument specifies the source, which can be source 1 or source 2. To set the actual waveform (channel, reference, or math) that is assigned to source, you must use the *MEASUREMENT:MEAS<x>:SOURCE<x> WFM* command. Two sources are provided because some measurements (gain, phase) require two sources.

---

**Group** Measurement

**Syntax** MEASUREMENT:MEAS<x>:SOURCE<x>:WFMDB:STATE { ON | OFF | NR1 }  
MEASUREMENT:MEAS<x>:SOURCE<x>:WFMDB:STATE?

**Related Commands** [MEASUREMENT:MEAS<x>:SOURCE<x>:WFMDB:SIGType](#)

- Arguments**
- ON enables the database measurement for the specified source waveform so that the measurement is made on the database.
  - OFF disables the database measurement so that the measurement is taken directly on the acquired waveform.
  - NR1 set to 0 disables the database measurement; any other value enables the database measurement.

**Examples** MEASUREMENT:MEAS5:SOURCE1:WFMDB:STATE ON causes Measurement 5 to be taken on the Source 1 waveform database.

MEASUREMENT:MEAS<x>:SOURCE<x>:WFMDB:STATE? might return  
MEASUREMENT:MEAS2:SOURCE1:WFMDB:STATE 0, indicating that  
Measurement 2 will be taken on the Source 1 waveform.

---

**NOTE.** A query always returns a 0 or 1.

---

## MEASUREMENT:MEAS<x>:STATE

This command sets or queries whether the specified measurement slot is computed and displayed. The measurement slot is specified by x, which can be 1 through 8. Sending this command is equivalent to selecting a measurement slot (such as Meas 3) and checking the On box in the Measurement Setup dialog box.

For a measurement to display, you must have selected a source waveform and defined the measurement you want take and display. You select the measurement source with the MEASUREMENT:MEAS<x>:SOURCE<x>:WFM command. You define the measurement type with the MEASUREMENT:MEAS<x>:TYPE command.

**Group** Measurement

**Syntax** MEASUREMENT:MEAS<x>:STATE {OFF | ON | <NR1>}  
MEASUREMENT:MEAS<x>:STATE?

**Related Commands** [MEASUREMENT:MEAS<x>:SOURCE<x>:WFM](#), [MEASUREMENT:MEAS<x>:TYPE](#)

- Arguments**
- OFF disables the calculation and display of the measurement defined in the specified measurement slot.
  - ON enables the calculation and display of the measurement defined in the specified measurement slot.
  - NR1 set to 0 disables the calculation and display of the specified measurement slot; any other value enables the calculation and display of the specified measurement slot.

**Examples** MEASUREMENT:MEAS1:STATE ON displays the measurement defined in measurement slot 1.

MEASUREMENT:MEAS4:STATE? might return MEASUREMENT:MEAS4:STATE 1, indicating that the measurement defined for measurement slot 4 is displayed.

---

**NOTE.** *A query always returns a 0 or 1.*

---

## MEASUREMENT:MEAS<x>:STATISTICS:CLEAR (No Query Form)

This command (no query form) clears and restarts the calculation of the statistics for the measurement slot specified by x, which can be 1 through 8.

**Group** Measurement

**Syntax** MEASUREMENT:MEAS<x>:STATISTICS:CLEAR

**Related Commands** [MEASUREMENT:STATISTICS:ENABLE](#)

**Examples** MEASUREMENT:MEAS3:STATISTICS:CLEAR clears and restarts calculating the statistics for Measurement 3.

## MEASUREMENT:MEAS<x>:STDDEV? (Query Only)

This query only command returns the standard deviation of values accumulated for the measurement slot specified by x, which can be 1 through 8, since the last statistical reset.

**Group** Measurement

**Syntax** MEASUREMENT:MEAS<x>:STDDEV?



**Related Commands**    [MEASUREMENT:MEAS<x>:ALL?](#)

**Returns**    NR3

**Examples**    MEASUREMENT:MEAS1:STDDEV? might return MEASUREMENT:MEAS1:STDDEV 1.16796169259E-011, indicating that the standard deviation for Measurement 1 is 11.68 ps.

## MEASUREMENT:MEAS<x>:TYPE

This command sets or queries the measurement type assigned to the specified measurement slot, where x is one of the available measurement slots from 1 through 8. If the specified measurement slot is already assigned, selecting a new measurement type will re-assign the specified slot to the requested measurement type. Sending this command is equivalent to using the pulldown list to select the type of measurement you want to take, after first selecting the measurement slot (such as Meas 1) and a source-signal type (such as NRZ), in the Meas Setup dialog box.

---

**NOTE.** All of the measurement types require at least one source waveform to be set with the `MEASUREMENT:MEAS<x>:SOURCE1:WFM` command.

---

You can set numerous measurement parameters that can enhance your measurement results and increase measurement repeatability, please refer to the Measurement Command Group for more information.

**Group**    Measurement

**Syntax**    MEASUREMENT:MEAS<x>:TYPE {UNDEFINED | HIGH | LOW | AMPLITUDE | MEAN | MAXIMUM | MINIMUM | PK2PK | MID | POVERSHOOT | NOVERSHOOT | RMSNOISE | PKPKNOISE | AOPTPWRDBM | RMS | ACRMS | CRMS | CMEAN | SNRATIO | AOPTPWR | GAIN | OMA | RISE | FALL | PERIOD | FREQUENCY | PCROSS | NCROSS | PWIDTH | NWIDTH | PDUTY | NDUTY | BURST | RMSJITTER | PKPKJITTER | DELAY | PHASE | AREA | CAREA | EXTINGCTDB | EXTINGCTPCT | EXTINGCTRATIO | EXTINGCTCAL | EYEHEIGHT | PCTCROSS | LEVCROSS | QFACTOR | EYEWIDTH | DISTDUTY | BITTIME | BITRATE | TIMCROSS | EYEOFACOR | SUPRSDB | SUPRSPCT | SUPRSRATIO | PULSESym}  
MEASUREMENT:MEAS<x>:TYPE?

## Arguments

Brief definitions of each measurement type follow. For a more comprehensive definition of a measurement type, based on the signal type of the measurement source, select Measurement Reference from the instrument Help menu.

For information on restricting a measurement to an area you specify, see the [MEASUREMENT:MEAS<x>:GATING:STATE](#) command.

- UNDEFINED deletes any measurement defined for Meas<x>, freeing it for use.
- HIGH measures the top (or Logic 1) reference level of the waveform.
- LOW measures the bottom (or Logic 0) level of the waveform.
- AMPLITUDE measures the difference between the High and Low level of the waveform.
- MEAN measures the arithmetic mean of the waveform within the measurement region.
- MAXIMUM measures the largest amplitude of the waveform within the measurement region.
- MINIMUM measures the smallest amplitude of the waveform within the measurement region. It is measured over the entire waveform or gated region.
- PK2PK measures the maximum amplitude divergence within the measurement region.
- MID measures the midpoint of the maximum amplitude divergence within the measurement region. The value is measured over the entire waveform or gated region.
- POVERSHOOT measures the percentage by which the maximum amplitude of the waveform exceeds the High level.
- NOVERSHOOT measures the Percentage by which the minimum amplitude of the waveform under-runs the Low level.
- RMSNOISE measures one standard deviation of the waveform amplitude variance. This measurement is taken within the eye aperture for NRZ and RZ waveforms or the center of the measurement region for Pulse waveforms.
- PKPKNOISE measures the maximum range of the waveform amplitude variance measured within either the eye aperture (for NRZ and RZ waveforms) or the center of the measurement region (Pulse waveforms).
- AOPTPWRDBM measures the true average power of an optical signal expressed in decibels referenced to 1 mw.
- RMS measures the true root mean square of the waveform within the measurement region.
- ACRMS measures the root mean square, minus the DC component, of the waveform within the measurement region.

- **CRMS** measures the root-mean-square voltage of the waveform within the first period of the measurement region.
- **CMEan** measures the arithmetic mean of the waveform over the first cycle of the measurement region.
- **SNRatio** measures the ratio of the amplitude of the signal to the noise within either the eye aperture (for NRZ and RZ waveforms) or the center of the measurement region (Pulse waveforms).
- **AOPTpwr** measures the True average power of an optical signal expressed in watts.
- **GAIN** measures the amplitude gain between two waveforms.
- **OMA** measures the optical modulation amplitude, either at the eye crossing (for NRZ waveforms) or at the specified center apertures of the high and low states (Pulse waveforms).
- **RISe** measures the time interval between the waveform crossing the low-reference and high-reference levels on a rising edge.
- **FALL** measures the time interval between the waveform crossing the high-reference and low-reference levels on a falling edge.
- **PERIOD** measures the time interval between two consecutive crossings on the same slope of the waveform at the mid-reference level.
- **FREQUENCY** measures the reciprocal of the Period of the waveform.
- **PCROSS** measures the time at which the waveform crosses the mid-reference level on a rising edge.
- **NCROSS** measures the time at which the waveform crosses the mid-reference level on a falling edge.
- **PWIdth** measures the time interval between consecutive rising and falling edges, measured at the mid-reference level.
- **NWIdth** measures the time interval between consecutive falling and rising edges, measured at the mid-reference level.
- **PDuty** measures the percentage of the waveform's +Width relative to its Period or Bit Time.
- **NDuty** measures the percentage of the waveform's -Width relative to its Period or Bit Time.
- **BURSt** measures the time between the first and last crossings, either positive or negative, of the waveform at the mid-reference level within the measurement region.
- **RMSJitter** measures the one standard deviation (s) of the time variance where the waveform crosses the mid-reference level.

- **PKPKJitter** measures the delta between the minimum and maximum of time crossings at the mid-reference level.
- **DELay** measures the time interval between the crossings of the mid-reference levels on the two sources of the measurement.
- **PHase** measures the Amount of lead or lag of one measurement source relative to a second measurement source. Phase is expressed in degrees, where  $360^\circ$  comprise one cycle or one bit of the waveform.
- **AREa** measures the area under the waveform curve within the measurement region.
- **CAREa** measures the area under the curve for the first bit or period of the waveform.
- **EXTINCTDB** measures the ratio of logic 1 to logic 0 expressed in decibels (dB).
- **EXTINCTPCT** measures the ratio of logic 0 to logic 1 expressed in percentage (%).
- **EXTINCTRATIO** measures the ratio of logic 1 to logic 0 expressed as a simple ratio.
- **EXTINCTCAL** precisely measures the ratio of logic 1 to logic 0 expressed in decibels (dB). The precision measurement is only available with the use an optical module with the Calibrated Extinction Ratio option.
- **EYEHeight** measures the vertical eye opening of the waveform guard-banded by  $\pm 3s$ , measured over the eye aperture.
- **EYEFactor** measures the horizontal eye opening of the waveform guard-banded by  $\pm 1s$ , measured over the eye aperture.
- **EYEWIdth** measures the horizontal eye opening of the waveform guard-banded by  $\pm 3s$ , measured at the mid-reference level.
- **PCTCROSS** measures the height of eye crossing as a percentage of eye height measured in the eye aperture.
- **LEVcROSS** measures the mean waveform amplitude at the eye crossing.
- **QFACTOR** measures the ratio of the amplitude of the eye to the total RMS noise on the High and Low levels.
- **DISTDuty** measures the ratio of the time interval between the points where the rising and the falling edges cross the mid-reference level and the bit time of the waveform.
- **BITTime** measures the time interval between the start of two successive data bits of the waveform.
- **BITRate** measures the inverse of the time interval between the start of two successive data bits of the waveform.

- **TIMCROSS** measures the horizontal position (measured from the trigger point) of the eye crossing.
- **SUPRSDB** measures the ratio of the average High level to the suppressed level measured between two consecutive RZ pulses expressed in decibels (dB).
- **SUPRSPCT** measures the Ratio of the average High level to the Suppressed level measured between two consecutive RZ pulses expressed in percentage (%).
- **SUPRSRATIO** measures the Ratio of the average High level to the Suppressed level measured between two consecutive RZ pulses expressed as a simple ratio.
- **PULSESym** measures the extent to which the RZ pulse is symmetrical around the peak at the mid-reference level. The pulse peak is the center of the interval, sized to Eye Aperture, which yields the maximum mean vertical value.

### Examples

**MEASUREMENT:MEAS3:TYPE FREQUENCY** defines Measurement 3 as a measurement of the frequency of a waveform.

**MEASUREMENT:MEAS3:TYPE?** might return **MEASUREMENT:MEAS3:TYPE RMS**, indicating that Measurement 3 is defined to measure the RMS value of a waveform.

## MEASUREMENT:MEAS<x>:UNITS? (Query Only)

This query only command returns the units for the measurement slot specified by x, which can be 1 through 8. You first must have defined the measurement slot using the **MEASUREMENT:MEAS<x>:TYPE** command before the units can be returned.

**Group** Measurement

**Syntax** **MEASUREMENT:MEAS<x>:UNITS?**

**Return Values.** **MEASUREMENT:MEAS<x>:UNITS? <QString>**

- "dB" for extinction ratio dB
- "Hz" for hertz
- "s" for seconds
- "unitless" for Q-factor
- "V" for volts
- "VV" for volts<sup>2</sup>
- "volt-seconds" for cycle area
- "W" for watts (used with optical modules)
- "watt-seconds" for cycle area
- "%" for percent
- "Ω" for ohms (used with TDR)
- "Ω -seconds" for cycle area
- "ρ" for rho (used with TDR)
- "ρ-seconds" for cycle area

**Related Commands**    [MEASUREMENT:MEAS<x>:VALUE?](#), [MEASUREMENT:MEAS<x>:TYPE](#)

**Examples**    MEASUREMENT:MEAS3:UNITS? might return MEASUREMENT:MEAS3:UNITS  
"%", indicating the units for Measurement 3 are percent.

## MEASUREMENT:MEAS<x>:VALUE? (Query Only)

This query only command returns the value that has been calculated for the measurement slot specified by x.

**Group**    Measurement

**Syntax**    MEASUREMENT:MEAS<x>:VALUE?

**Related Commands**    [MEASUREMENT:MEAS<x>:TYPE](#), [MEASUREMENT:MEAS<x>:UNITS?](#)

**Returns**    NR3 is the calculated value for the specified measurement.

**Examples** MEASUREMENT:MEAS1:VALUE? might return MEASUREMENT:MEAS1:VALUE 1.24896277383E+008 as the calculated value for Measurement 1.

## MEASUREMENT:STATISTICS:ENABLE

This command sets or queries whether the instrument calculates the measurement statistics. Sending this command is equivalent to selecting the Statistics check box in the Measurement Setup dialog box.

Use the MEASUREMENT:STATISTICS:WEIGHTING command to set statistics weighting. Weighting is the number of measurements over which you want the statistics calculated.

Use the MEASUREMENT:MEAS<x>:ALL? Command to query all the measurement statistical values at one time.

**Group** Measurement

**Syntax** MEASUREMENT:STATISTICS:ENABLE { ON | OFF | NR1 }  
MEASUREMENT:STATISTICS:ENABLE?

**Related Commands** [MEASUREMENT:STATISTICS:WEIGHTING](#)  
[MEASUREMENT:MEAS<x>:ALL?](#)

- Arguments**
- ON causes the instrument to compute the statistics on measurement values. If a measurement is using a waveform database, it will not compute statistics.
  - OFF turns off the calculation of the measurement statistics.
  - NR1 set to 0 disables the calculation of statistics; any other value enables the calculation of statistics.

**Examples** MEASUREMENT:STATISTICS:ENABLE ON enables the calculation of the measurement statistics.

MEASUREMENT:STATISTICS:ENABLE? might return MEASUREMENT:STATISTICS:ENABLE 0, indicating that the calculation of statistics is disabled.

---

**NOTE.** A query always returns a 0 or 1.

---

## MEASUREMENT:STATISTICS:WEIGHTING

This command sets or queries the weighting the instrument uses to calculate the measurement statistics. Weighting is the number of measurements over which you want the statistics calculated. Sending this command is equivalent to entering a value in the Weighting control in the Measurements Setup dialog box. The Statistics check box must be selected before the Weighting control is active.

Use the MEASUREMENT:STATISTICS:ENABLE command to enable the statistics weighting function.

**Group** Measurement

**Syntax** MEASUREMENT:STATISTICS:WEIGHTING <NR1>  
MEASUREMENT:STATISTICS:WEIGHTING?

**Related Commands** [MEASUREMENT:STATISTICS:ENABLE](#)

**Arguments** NR1 is the number of measurements over which the statistics are calculated. The range is from 2 through 1,000,000 with a resolution of 1. The default value is 32.

**Examples** MEASUREMENT:STATISTICS:WEIGHTING 10 sets the weighting to 10, which means the statistics are calculated over 10 measurements.

MEASUREMENT:STATISTICS:WEIGHTING? might return  
MEASUREMENT:STATISTICS:WEIGHTING 50, indicating that the statistics are calculated over 50 measurements.

## \*OPC

This command 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. For a complete discussion of the use of these registers and the output queue, see Registers and Queues.

The \*OPC command allows you to synchronize the operation of the instrument with your application program. For more information, see Synchronization Methods.



**Commands that Generate an OPC Message.**

Operation	Command
Automatic instrument adjustment	<a href="#">AUTOSet EXECute</a>
Internal self compensation	<a href="#">COMPensate:ALLModules</a> <a href="#">COMPensate:CH&lt;x&gt;</a> <a href="#">COMPensate:DARKLev:CH&lt;x&gt;</a> <a href="#">COMPensate:MAInframe</a> <a href="#">COMPensate:OPTGAIN:CH&lt;x&gt;</a>
Conditional	<a href="#">ACQUIRE:STATE ON</a> or <a href="#">ACQUIRE:STATE RUN</a> (when <a href="#">ACQUIRE:STOPAfter?</a> is set to <a href="#">CONDition</a> )
Hardcopy output	<a href="#">HARDCopy START</a>
Phase reference	<a href="#">PHAseref:CHAR</a>

**Group** Status and Error

**Syntax** \*OPC  
\*OPC?

**Related Commands** [BUSY?](#), [\\*WAI](#)

**Examples** \*OPC generates the operation complete message in the SESR at the completion of all pending OPC operations.  
\*OPC ? might return 1, indicating that all pending OPC operations are finished.

**\*OPT? (Query Only)**

This query-only command returns a list of the options installed in your instrument.

**Group** Status and Error

**Syntax** \*OPT?

**Examples** \*OPT? might return "DSA8300: Advanced Trigger with Pattern Sync".

## PHAseref:CHAR (No Query Form)

This command runs a phase reference characterization on the clock connected to the input of the Phase Reference Module that is currently selected as the Phase Ref source.

After connecting a clock to the Phase Reference Module, turning the module on (using the PHAseref:MODE command), and notifying the instrument of its frequency (using the PHAseref:<CHx>:FREQUENCY command), use the PHAseref:CHAR command to characterize the clock signal.

---

**NOTE.** Use the *PHAseref:CH<x>:STATUS?* command to ascertain the outcome of the characterization. There is no query form for *PHAseref:CHAR*.

---

**Group** Phase Reference

**Syntax** PHAseref:CHAR

**Related Commands** [PHAseref:CH<x>:FREQUENCY](#), [PHAseref:MODE](#), [PHAseref:SOURce](#), [PHAseref:CH<x>:RANge?](#), [PHAseref:CH<x>:STATUS?](#)

**Examples** PHASEREF:CHAR initiates a characterization of the currently selected Phase Reference Module.

## PHAseref:CH<x>:FREQUENCY

This command sets (notifies the instrument of) or queries the frequency of the phase reference clock that is connected to the Phase Reference Module specified by CH <x>, where <x> is a channel associated with the compartment in which the Phase Ref Module is installed. You can use either of the two channels associated with the compartment. For example, if the module is installed in the CH7/CH8 compartment, x can be 7 or 8.

---

**NOTE.** You must connect a phase reference clock to the input connector of the Phase Reference Module to use the phase-correction timebase that it supplies. Then, you must specify to the instrument the source, using the *PHAseref:SOURce* command, and the frequency of the clock that you have supplied, using the *PHAseref:CH<x>:FREQUENCY* command.

---

**Group** Phase Reference

**Syntax** PHAseref:CH<x>:FREQUency <NR3>  
PHAseref:CH<x>:FREQUency?

**Related Commands** PHAseref:CHAR, PHAseref:MODE, PHAseref:SOUrce, PHAseref:CH<x>:RANge?, PHAseref:CH<x>:STAtus?

**Arguments** <NR3> specifies the phase reference clock to the instrument. Set the value that matches the clock that you have connected to the Phase Reference Module. Values you specify that mismatch the input clock frequency by 10% or more are likely to result in an error during characterization...

The instrument accepts an allowed range for <NR3> from 2E+09 to 25E+09 for the 82A04 or from 2E+09 to 110E+09 for the 82A04-60G. Values outside the allowed range are coerced to the minimum or maximum of the range.

**Options F1 and F2.** These options are exceptions to the preceding note. Options F1 and F2 provide filters and cabling that, when used to connect the phase reference clock, guarantee operation at certain frequencies below 8 GHz, the normal minimum limit that's guaranteed. If you've connected these options, enter a <NR3> argument that matches the lower clock frequency that you have connected. See your *DSA8300 Specifications and Performance Verification Manual* for more information on the 82A04 module specifications.

**Examples** PHASEREF:CH2:FREQUENCY 43.01841400000E+9 sets the phase reference clock frequency to match a 43.018414 GHz clock connected to the 82A04-60G Phase Reference Module installed in the C1C2 small module compartment.

This same command will result in a coerced clock frequency of 25 GHz for 82A04 modules that are not equipped with Option 60G. With the connected clock at 45 GHz, turning phase correction on and performing a characterization results in an error, since the coerced setting mismatches the connected clock by more than 10%.

## PHAseref:CH<x>:RANge? (Query Only)

This command queries and returns the guaranteed frequency range of the Phase Reference Module corresponding to CH<x>, where <x> is a channel associated with the compartment in which the Phase Ref Module is installed. You can use either of the two channels associated with the compartment. For example, if the module is installed in the CH 7/CH8 compartment, x can be 7 or 8.

The guaranteed operation range of the Phase Reference Module is:

- Standard 82A04: from 8E+09 to 25E+09
- Option 60G-equipped 82A04: from 8E+09 to 60E+09

---

**NOTE.** *The allowed range of values that is accepted is larger than the guaranteed range that's supported. See the [PHAseref:CH<x>:FREQuency](#) command for more information.*

---

**Group** Phase Reference

**Syntax** PHAseref:CH<x>:RANge?

**Related Commands** [PHAseref:CH<x>:FREQuency](#), [PHAseref:CHAR](#), [PHAseref:MODE](#), [PHAseref:SOUrce](#), [PHAseref:CH<x>:STAtus?](#)

**Examples** PHASEREF:CH2:RANGE? might return PHASEREF:CH2:RANGE  
8.000GHZ-60.00GHZ for an 82A04 module equipped with Option 60G and that is installed in the C1C2 small module compartment.

## PHAseref:CH<x>:STAtus? (Query Only)

This command queries and returns the results of the most recent phase reference characterization of the Phase Reference Module specified by CH <x>, where <x> is a channel associated with the compartment in which the Phase Ref Module is installed. You can use either of the two channels associated with the compartment. For example, if the module is installed in the CH 7/CH8 compartment, x can be 7 or 8. The string returned is one of the following:

**Characterization passed.** This status indicates that the clock that you've input has been successfully characterized and has since been stable.

**Characterization required.** This status indicates that you need to characterize the clock connected to the currently selected and enabled Phase Reference module. Either you have not run a successful characterization since you connected the clock and turned phase correction on, or you have changed the clock input, or the clock has changed in amplitude, offset or shape, or you have changed one or more of the following settings:

- Changed the Phase Ref mode (except when setting to Off)
- Recalled a setup
- Recalled the Factory/Default setup
- Compensated the main instrument
- Recalled a main instrument compensation
- Changed the Trigger level\*
- Changed the Trigger slope\*
- Changed the Trigger source\*
- Changed the Clock recovery source when trigger source is clock recovery\*
- Changed the Clock recovery rate, enum or user adjustable, for the selected clock recovery channel when trigger source is clock recovery\*

---

**NOTE.** *Changes to these trigger controls only cause Required status when made in Triggered Phase Ref Mode.*

---

- Changed Phase reference input frequency
- Compensated the Phase Reference Module
- Recalled a Phase Reference Module compensation

In any of these cases, make sure you have a stable, valid clock connected, and then use the PHAseref:CHAR command to run characterization.

**Characterization failed.** This status indicates that characterization failed. Verify the clock signal and rerun characterization. If fail status still results, you may be providing a faulty clock or one that mismatches the clock frequency that you set with the PHAseref:CH<x>:FREQuency command. Here are some possible status messages that return with the "Characterization failed" string:

- Acq System Error
- Entered Frequency Incorrect
- Frequency too Low
- No Phase Ref Clock
- No Triggers
- Phase Ref Clock Amplitude Low
- Phase Ref Clock Amplitude High
- Reference Clock Shape Problem
- Signal to Noise Ratio Low

**Group** Phase Reference

**Syntax** PHAseref:CH<x>:StAtus?

**Related Commands** [PHAseref:CH<x>:FREQuency](#), [PHAseref:CHAR](#), [PHAseref:MODe](#), [PHAseref:SOUrce](#), [PHAseref:CH<x>:RANge?](#)

**Examples** PHASEREF:CH1:STATUS? might return PHASEREF:CH1:STATUS CHARACTERIZATION REQUIRED, indicating that you need to characterize the clock connected to the currently selected and enabled Phase reference module.

## PHAseref:MODe

Sets or queries the phase correction mode, Off, Freerun, Triggered, or SSC Triggered for the currently selected Phase Reference Module (if any).

---

**NOTE.** Characterization (see the [PHAseref:CHAR](#) command) is required whenever the phase correction mode is changed.

---

The 82A04 Phase Reference Module provides a phase-correction timebase. This timebase, in turn, supports ultra-low trigger jitter (typical is 200 fsec RMS), improving the fidelity of acquired signals.

---

**NOTE.** Data acquisition is fundamentally different when the instrument acquires data using the phase-correction timebase versus the internal timebase. To investigate phase-correction-timebase and internal-timebase behaviors relative to acquisition, see the instrument online help.

---

<b>Group</b>	Phase Reference
<b>Syntax</b>	PHASeref:MODE {OFF   FREerun   TRIGger   SSCTrigger} PHASeref:MODE?
<b>Related Commands</b>	PHASeref:CH<x>:FREQuency, PHASeref:CHAR, PHASeref:MODE, PHASeref:SOUrce, PHASeref:CH<x>:RANge?, PHASeref:CH<x>:STAtus?
<b>Arguments</b>	<ul style="list-style-type: none"> <li>■ OFF turns off the phase correction timebase, so that the instrument uses its normal timebase.</li> <li>■ FREerun turns on the phase correction timebase and runs it in Free Run mode, in which the trigger is ignored for purposes of determining the horizontal acquisition window.</li> <li>■ TRIGger turns on the phase correction timebase in Triggered mode in which the trigger event is used to determine horizontal acquisition window. The trigger that you provide determines where in the data stream the waveform record is taken.</li> <li>■ SSCTrigger turns on the phase correction timebase in Triggered Spread Spectrum Clock mode. Triggered (SSC) Mode operates exactly the same as Triggered Mode except that the instrument assumes, and accommodates (up to approximately 5000 ppm), some amount of Spread Spectrum Clocking contained on the Phase Reference input clock.</li> </ul>
<b>Examples</b>	<p>PHASEREF:MODE TRIGGER enables the phase correction timebase in the Triggered mode, using the Phase Reference Module set as the phase reference source (selectable using the PHASeref:SOUrce command).</p> <p>PHASEREF:MODE? might return PHASEREF:MODE OFF , indicating that the instrument is using its normal timebase.</p>

## PHASeref:SOUrce

This command sets or queries the small-module compartment that is used as the phase ref source when phase-correction is turned on. The compartment must contain a Phase Reference module. If only one such module is installed, the instrument defaults to the source matching the compartment in which it is installed.

---

**NOTE.** *The 82A04 Phase Reference Module can only be installed in one of the four small-module compartments. A large module installed in the large C1C2 compartment disables the small C1C2 compartment; a large module in the large C3C4 compartment disables the small C3C4 compartment. Under these conditions, any Phase Reference Module installed in the disabled compartments cannot be used.*

---

**Group** Phase Reference

**Syntax** PHAseref:SOURCE { C1C2 | C3C4 | C5C6 | C7C8 }  
PHAseref:SOURCE?

**Related Commands** PHAseref:CH<x>:FREQuency, PHAseref:CHAR, PHAseref:MODE,  
PHAseref:CH<x>:RANge?, PHAseref:CH<x>:STATus?

- Arguments**
- C1C2 selects the first (left-most) small-module compartment as the Phase Reference Module source. This argument is ineffective if a large module is installed in the first (left-most) large module compartment (see note above).
  - C3C4 selects the second small-module compartment as the Phase Reference Module source. This argument is ineffective if a large module is installed in the second large module compartment (see the note above).
  - C5C6 selects the third small-module compartment as the Phase Reference Module source.
  - C7C8 selects the fourth (right-most) small-module compartment as the Phase Reference Module source.

**Examples** PHASEREF:SOURCE C5C6 sets the instrument to use the Phase Reference Module in the small-module compartment for channels 5 and 6 as the source for the phase correction timebase.

PHASEREF:SOURCE? might return PHASEREF:SOURCE C1C2, indicating that the currently selected source is the Phase Reference Module that is installed in the small-module compartment for channels 1 and 2.

## \*PSC

This command sets or queries the power-on status flag that controls the automatic power-on handling of the DESER, SRER, and ESER registers. When \*PSC is true, the DESER register is set to 255 and the SRER and ESER registers are set to 0 at power on. When \*PSC is false, the current values in the DESER, SRER,



and ESER registers are retained in nonvolatile memory when the instrument is powered off and are restored when the instrument is powered on.

<b>Group</b>	Miscellaneous
<b>Syntax</b>	*PSC <NR1> *PSC?
<b>Arguments</b>	NR1 set to 0 sets the power-on status clear flag to false. When the argument is not equal to 0, the power-on status clear flag is set to true.
<b>Examples</b>	*PSC 0 sets the power-on status flag to false, causing the values in the DESER, SRER, and ESER registers to be retained in nonvolatile memory when the instrument is powered off and restored when it is powered on.  *PSC? might return 1, indicating that the power-on status clear flag is set to true.

## RECALL:SETUp (No Query Form)

This command (no query form) restores a stored or factory default setup from a Windows file. Sending this command is equivalent to selecting Recall Saved Setup or Recall Factory Setup in the Save/Recall Setup menu.

<b>Group</b>	Save and Recall
<b>Syntax</b>	RECALL:SETUp {FACTory   <file path>}
<b>Related Commands</b>	<a href="#">FACTory</a> , <a href="#">*RST</a> , <a href="#">SAVE:SETUp</a>
<b>Arguments</b>	<ul style="list-style-type: none"> <li>■ FACTory selects the factory setup.</li> <li>■ &lt;file path&gt; is the location from which the setup will be recalled.</li> </ul> <p>The &lt;file path&gt; is a quoted string that defines the file name and path. Input the file path using the form &lt;drive&gt;\&lt;dir&gt;\&lt;filename&gt;. &lt;drive&gt; and one or more &lt;dir&gt;s are optional. If you do not specify them, the instrument will read the file from the default directory. The &lt;filename&gt; can be a Windows long file name. Do not use wild card characters.</p>
<b>Examples</b>	RECALL:SETUP FACTORY recalls (and makes current) the instrument setup to its factory defaults.

RECALL:SETUP "TEK00000.STP" recalls the instrument setup from the file TEK00000.STP in the default directory and on the default drive.

## RECALL:WAVEform (No Query Form)

This command (no query form) recalls a stored waveform into a reference location. Sending this command is equivalent to selecting Recall Waveform from the File Menu, and then selecting a waveform file source and reference destination in the Recall Waveform dialog box.

**Group** Save and Recall

**Syntax** RECALL:WAVEform <file path>,REF<x>

- Arguments**
- <filepath> is a quoted string that defines the file name and path. Input the file path using the form <drive>/<dir>/<filename>. <drive> and one or more <dir>s are optional. If you do not specify them, the instrument will recall the waveform from the default directory. The <filename> can be a Windows long file name. Do not use wild card characters.
  - REF<x> is the location in internal reference memory to which the waveform is recalled.

**Examples** RECALL:WAVEFORM "TEK00000.WFM",REF1 recalls the waveform stored in the file named TEK00000.WFM in the default directory and on the default drive to Reference waveform 1.

## REF<x>:POSition

This command allows you to set or query the vertical position of the specified reference waveform. The reference waveform is specified by x, which can be 1 through 8. This is the equivalent to specifying a reference waveform in the Waveform section of the Vertical Setup dialog box and then entering a value for Position in the Setup section.

**Group** Vertical

**Syntax** REF<x>:POSITION <NR3>  
REF<x>:POSITION?

<b>Arguments</b>	NR3 is the desired position, in divisions from the center graticule. The range is $\pm 1000$ divisions.
<b>Examples</b>	<p>REF2:POSITION 1.3E+000 positions the Reference 2 waveform 1.3 divisions above the center of the display.</p> <p>REF1:POSITION? might return REF1:POSITION 2.000000000000, indicating that the Reference 1 waveform is positioned 2 divisions above the center of the display.</p>

## REF<x>:SCALE

This command allows you to set or query the vertical scale of the specified reference waveform. The waveform is specified by x, which can be 1 through 8. This is the equivalent to specifying a reference waveform in the Waveform section of the Vertical Setup dialog box and then entering a value for Scale in the Setup section.

Increasing the Scale causes the waveform to be displayed smaller. Decreasing the scale causes the waveform to be displayed larger. For reference waveforms, this setting controls the display only, graphically scaling these waveforms and having no effect on the acquisition hardware.

<b>Group</b>	Vertical
<b>Syntax</b>	<pre>REF&lt;x&gt;:SCALE &lt;NR3&gt; REF&lt;x&gt;:SCALE?</pre>
<b>Arguments</b>	NR3 is the vertical reference scale in units per division.
<b>Examples</b>	<p>REF4:SCALE 1.0E-02 sets the Reference 4 waveform scale to 10 mV per division.</p> <p>REF1:SCALE? might return REF1:SCALE 0.100000000000, indicating that the current volts per division setting of the Reference 1 waveform is 100 mV per division.</p>

## REF<x>:WFMLabel

This command sets or queries the label associated with the reference waveform specified.

<b>Group</b>	Vertical
<b>Syntax</b>	REF<x>:WFMLabel <Qstring> REF<x>:WFMLabel?
<b>Related Commands</b>	<a href="#">CH&lt;x&gt;:WFMLabel</a> , <a href="#">MATH&lt;x&gt;:WFMLabel</a>
<b>Arguments</b>	Qstring sets the label for the reference waveform.
<b>Examples</b>	REF1:WFMLABEL "MY REF1 WAVEFORM" sets the label for REF1 to be "My REF1 waveform," which is the label displayed with the waveform when it is displayed on screen.  REF1:WFMLABEL? might return REF1:WFMLABEL "MY REF1 WAVEFORM", indicating that the label for the REF1 waveform is set to "My REF1 waveform."

## \*RST (No Query Form)

This command (no query form) resets the instrument to the factory default settings.

The \*RST command does not alter the following:

- The state of the IEEE Std 488.1-1987 interface
- The selected IEEE Std 488.1-1987 address of the instrument
- Calibration data that affect device specifications
- The Output Queue
- The Service Request Enable Register setting
- The Standard Event Status Enable Register setting
- The Power-on status clear flag setting
- Stored settings

This command is equivalent to pressing the **DEFAULT SETUP** button on the front panel.

<b>Group</b>	Status and Error
<b>Syntax</b>	*RST
<b>Related Commands</b>	<a href="#">FACTory</a> , <a href="#">RECALL:SETUp</a> , <a href="#">SAVE:SETUp</a>

**Examples** \*RST resets the instrument settings to factory defaults.

## SAVe:SETUp (No Query Form)

This command (no query form) saves the current instrument setup into the specified memory location. Sending this is equivalent to selecting Save Setup in the File menu.

**Group** Save and Recall

**Syntax** SAvE:SETUp <file path>

**Related Commands** [FACTory](#), [RECAll:SETUp](#)

**Arguments** <FILE PATH> is a quoted string that defines the file name and path. Input the file path using the form <drive>/<dir>/<filename>. <drive> and one or more <dir>s are optional. If you do not specify them, the instrument will write the file to the current directory. The <filename> can be a Windows long file name. Do not use wild card characters. It is recommended that you use "STP" for the extension to identify files that store setup data.

Settings saved in one instrument may or may not work on a different instrument with a different version of firmware.

**Examples** SAvE:SETUP "C:\MY DOCUMENTS\TESTS\UI\DATA\SETUPTEST.STP" saves the current instrument setup to the file Setuptest.STP in the Data subdirectory on the C drive.

## SAVe:WAVEform (No Query Form)

This command (no query form) stores a waveform in one of eight reference memory locations or to a file. You must specify a waveform, a timebase, and a destination (either a file path or a waveform reference slot). Sending this command is equivalent to selecting Save Waveform... from the File menu and then selecting a waveform and destination in the Save Waveform(s) dialog box.

**Group** Save and Recall

**Syntax** SAvE:WAVEform {CH<x> | MATH<x> | REF<x>}, {MAIn | MAG1 | MAG2}, {REF<x> | <filepath>}

**Related Commands**    [DELEte:WAVEform](#)

- Arguments**
- CH<x> selects a channel waveform to save. The range for x is 1 through 8.
  - MATH<x> selects a math waveform to save. The range for x is 1 through 8.
  - REF<x> selects a reference waveform to save. The range for x is 1 through 8. You can also use this argument as a destination into which a waveform (including a reference waveform) can be saved.
  - MAIn selects Main as the timebase for the saved waveform. This is the default timebase.
  - MAG1 selects MAG1 as the timebase for the saved waveform.
  - MAG2 selects MAG2 as the timebase for the saved waveform.
  - <file path> is the location to which the waveform will be saved.

The <file path> is a quoted string that defines the file name and path. Input the file path using the form <drive>/<dir>/<filename>. The <filename> can be a Windows long file name. Do not use wild card characters.

**Examples**    SAVE:WAVEFORM MATH2,MAIN,REF1 saves the Math 2 waveform displayed by the Main timebase in Reference memory location 2.

SAVE:WAVEFORM MATH1,MAG1,"TEK00000.WFM" saves the Math1 waveform displayed by the Mag 1 timebase to the file TEK00000.WFM in the default directory and on the default drive.

## SElect? (Query Only)

This is a query only command that returns the selected waveform that is affected by the front-panel controls and the display status (on or off) of all waveforms.

**Group**    Vertical

**Syntax**    SElect?

**Examples**    SELECT? might return <PARA>SELECT:CH1 0;CH2 0;CH3 1;CH4 0;CH5 0;CH6 0;CH7 0;CH8 0;MATH1 1;MATH2 0;MATH3 0;MATH4 0;MATH5 0;MATH6 0;MATH7 0;MATH8 0;REF1 1;REF2 0;REF3 0;REF4 0;REF5 0;REF6 0;REF7 0;REF8 0;CONTROL REF1,MAIN</PARA>.

## SElect:CH<x>

This command sets or queries whether the specified acquisition waveform is displayed. The waveform is specified by x, which can be 1 through 8. This command is the equivalent to selecting an acquisition waveform and checking the On control in the Waveform section of the Vertical Setup dialog box.

**Group** Vertical

**Syntax** SElect:CH<x> { ON | OFF | NR1 }  
SElect:CH<x>?

**Related Commands** [SElect:CONTROL](#)

- Arguments**
- ON displays the specified acquisition waveform.
  - OFF turns off the display of the specified acquisition waveform.
  - NR1 set to 0 turns off the display of the specified acquisition waveform; any other value displays the specified acquisition waveform.

**Examples** SELECT:CH4 1 displays the acquisition waveform for Channel 4.  
SELECT:CH2? might return SELECT:CH2 0, indicating that the display of the Channel 2 acquisition waveform is disabled.

---

**NOTE.** A query always returns a 0 or 1.

---

## SElect:CONTROL

This command sets or queries the waveform that is currently affected by the front-panel controls. This is the equivalent to pressing the Vertical CH, MATH, or REF button on the front panel and then pressing a button (1 through 8) to specify a waveform.

When sending this command, you must specify a timebase (MAIn, MAG1, or MAG2). Also, the waveform must be displayed for the command to enable control of it.

---

**NOTE.** You should define a math waveform before turning the waveform on.

---

**Group** Vertical

**Syntax** `SElect:CONTROL {CH<x> | MATH<x> | REF<x> | NONE}, {MAIn | MAG1 | MAG2}`  
`SElect:CONTROL?`

**Related Commands** [SElect:CH<x>](#)

- Arguments**
- CH<x> selects the specified channel waveform as the waveform that is affected by the front-panel controls. The range for x is 1 through 8.
  - MATH<x> selects the specified math waveform as the waveform that is affected by the front-panel controls. The range for x is 1 through 8.
  - REF<x> selects the specified reference waveform as the waveform that is affected by the front-panel controls. The range for x is 1 through 8.
  - NONE causes the instrument to do nothing. When returned in a query, NONE indicates that no waveforms are associated with the front-panel controls.
  - MAIn selects Main as the timebase for the specified waveform.
  - MAG1 selects MAG1 as the timebase for the specified waveform.
  - MAG2 selects MAG2 as the timebase for the specified waveform.

**Examples** `SElect:CONTROL CH1,MAIn` selects the Channel 1 acquisition waveform as the waveform affected by the front-panel controls. This waveform is displayed using the main timebase.

`SElect:CONTROL?` might return `SElect:CONTROL math5,maIn`, indicating that the Math 5 waveform is currently affected by the front-panel controls and is displayed using the main timebase.

`SElect:CONTROL?` might return `SElect:CONTROL NONE,MAIn`, indicating that no waveform using the Main timebase is controlled by front-panel controls.

## SElect:MATH<x>

This command sets or queries whether the specified math waveform is displayed. The waveform is specified by x, which can be 1 through 8. This command is the equivalent to selecting a math waveform and checking the On control in the Waveform section of the Vertical Setup dialog box.

**Group** Vertical

**Syntax** `SElect:MATH<x> { ON | OFF | NR1 }`  
`SElect:MATH<x>?`



**Related Commands**    [SElect:CONTROL](#)

- Arguments**
- ON displays the specified math waveform.
  - OFF turns off the display of the specified math waveform.
  - NR1 set to 0 turns off the display of the specified math waveform; any other value displays the specified math waveform.

**Examples**    SELECT:MATH4 1 displays the Math 4 waveform.  
 SELECT:MATH2? might return SELECT:MATH2 0, indicating that the display of the Math 2 waveform is disabled.

**NOTE.** *A query always returns a 0 or 1.*

## SElect:REF<x>

This command sets or queries whether the specified reference waveform is displayed. The waveform is specified by x, which can be 1 through 8. This command is the equivalent to selecting a reference waveform and checking the On control in the Waveform section of the Vertical Setup dialog box.

**NOTE.** *You should define a reference waveform before turning the waveform on.*

**Group**    Vertical

**Syntax**    SElect:REF<x> { ON | OFF | NR1 }  
 SElect:REF<x>?

**Related Commands**    [SElect:CONTROL](#)

- Arguments**
- ON displays the specified reference waveform.
  - OFF turns off the display of the specified reference waveform.
  - NR1 set to 0 turns off the display of the specified reference waveform; any other value displays the specified reference waveform.

**Examples**    SELECT:REF4 1 displays the Reference 4 waveform.  
 SELECT:REF5? might return SELECT:REF5 0, indicating that the display of the Reference 5 waveform is disabled.

---

**NOTE.** *A query always returns a 0 or 1.*

---

## SET? (Query Only)

This query only command returns a string listing the instrument settings, except for configuration information for the calibration values. You can use this string to return the instrument to the state it was in when you made the SET? query. This command is identical to the \*LRN? command.

**Group** Miscellaneous

**Syntax** SET?

---

**NOTE.** *The SET? query always returns a string with command headers, regardless of the setting of the HEADer command. This is because the returned string is intended to be sent back to the instrument as a command string. The VERBose command can still be used to specify whether the returned headers should be abbreviated or full length.*

---

**Related Commands** [HEADer](#), [\\*LRN?](#), [VERBose](#)

**Examples** SET? might return a string (partial) that looks like this:

```
ACQUIRE:MODE SAMPLE; NUMAVG 16; SAVEFILE:SAVESCREEN
""; SAVEWFM """; ACQUIRE:STATE 1; STOPAFTER:MODE
RUNSTOP; BELL 0; ACTION NONE; CONDITION ACQWFMS;
COUNT 1; :CALIBRATE:DCCALIBRATOR 0.0000; :DIAG:LOOP
1; CONTROL:LOOP 0; HALT 0; :CURSOR:CURSOR1:COLOR 5;
SOURCE CH1,MAIN; :CURSOR:CURSOR2:COLOR 5; SOURCE
CH1,MAIN; :CURSOR:DIV2 0; FUNCTION OFF; SELECT
CURSOR1; HBARS:POSITION1 -0.00162162000; POSITION2
0.00170271000; :CURSOR:VBARS:POSITION1 1.67800000000E-6;
POSITION2 3.32800000000E-6; CURSOR:WAVEFORM:POSITION1
1.67800000000E-6; POSITION2 3.32800000000E-6;
:DATA:DESTINATION REF1; SOURCE CH1; ENCDG ASCII;
START 1;STOP 500; :WFMINPRE:ENCDG ASCII; BN_FMT RI;
BYT_OR LSB;NR_PT 500;PT_FMT Y;XINCR 0.0000; XZERO
16.00000000000E-9;XUNIT """;YMULT 0.10000000000; XMULT
100.00000000000E-12; XOFF 16.00000000000E-9; YZERO 0.0000;
YSCALE 0.0000; YOFF 0; ...
```

**\*SRE**

The \*SRE (Service Request Enable) command sets or queries the bits in the Service Request Enable Register (SRER). For more information, refer to Registers.

**Group** Status and Error

**Syntax** \*SRE <NR1>  
\*SRE?

**Related Commands** [\\*CLS](#), [DESE](#), [\\*ESE](#), [\\*ESR?](#), [EVENT?](#), [EVMsg?](#), [FACTory](#), [\\*STB?](#)

**Arguments** 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. The power-on default for SRER is 0 if \*PSC is 1. If \*PSC is 0, the SRER maintains its value throughout a power cycle.

**Examples** \*SRE 48 sets the bits in the SRER to 00110000 binary.  
\*SRE? returns a value of 32, showing that the bits in the SRER have the binary value 00100000.

**\*STB? (Query Only)**

The \*STB? (Read Status Byte) query returns the contents of the Status Byte Register (SBR) using the Master Summary Status (MSS) bit. For more information, refer to Registers.

**Group** Status and Error

**Syntax** \*STB?

**Related Commands** [\\*CLS](#), [DESE](#), [\\*ESE](#), [\\*ESR?](#), [EVENT?](#), [EVMsg?](#), [FACTory](#), [\\*SRE](#)

**Returns** <NR1>

**Examples** \*STB? might return 96 showing that the SBR contains the binary value 01100000.

## SYNC:TIMEOUT

This command sets or queries the default synchronization timeout for commands that rely upon data production by the instrument. The default timeout value of 3 seconds is appropriate for most applications, but you might need to adjust this timeout under the following circumstances:

- Slow waveform acquisition rate. The combination of record-length, averaging, frame-scan, trigger hold-off and trigger-source repetitive-rate settings can result in a waveform acquisition time that exceeds the synchronization timeout. If this is the case, sending a query to the instrument results in a synchronization timeout before the instrument can produce results.
- Custom user configurations. Any configuration that significantly influences the rate at which the instrument produces results can result in a synchronization timeout. Such configurations can include any third-party software that you install if it consumes a significant amount of CPU cycles or system resources.

**Group**      Miscellaneous

**Syntax**      SYNC:TIMEOUT <NR1>  
 SYNC:TIMEOUT?

**Related Commands**      MEASUREMENT:ALL:VALUE?, MEASUREMENT:LIST, MEASUREMENT:LISTVALUE?, MEASUREMENT:MEAS<x>:MAXIMUM?, MEASUREMENT:MEAS<x>:MINIMUM?, MEASUREMENT:MEAS<x>:MEAN?, MEASUREMENT:MEAS<x>:STDDEV?, MEASUREMENT:MEAS<x>:VALUE?, MEASUREMENT:MEAS<x>:ALL?, MEASUREMENT:MEAS<x>:UNITS?, MEASUREMENT:MEAS<x>:TYPE

ACQUIRE:CURRENTCOUNT:ACQWFMS?, ACQUIRE:CURRENTCOUNT:HISTWFMS?, ACQUIRE:CURRENTCOUNT:HISTHITS?, ACQUIRE:CURRENTCOUNT:MASKWFMS?, ACQUIRE:CURRENTCOUNT:MASKSAMPLES?, ACQUIRE:CURRENTCOUNT:MASKTHITS?, ACQUIRE:CURRENTCOUNT:MASKHITS<x>?, MASK:COUNT:TOTAL?, MASK:COUNT:WAVEFORMS?, MASK:COUNT:SAMPLES?, MASK:MASK<x>:COUNT?, HISTOGRAM:STATISTICS?, HISTOGRAM:STATISTICS:MEAN?, HISTOGRAM:STATISTICS:MEDIAN?, HISTOGRAM:STATISTICS:STDDEV?, HISTOGRAM:STATISTICS:PKTOPK?, HISTOGRAM:STATISTICS:SIGMA<x>?, HISTOGRAM:STATISTICS:PEAKHITS?, HISTOGRAM:STATISTICS:HITS?, HISTOGRAM:STATISTICS:WAVEFORMS?

**Arguments**      NR1 is the desired timeout duration specified in seconds. When any of the related commands are issued to the instrument, if query results are not produced before the timeout expires, a timeout condition will occur. Typically, this will result in error event being posted to the Event Queue and will be reflected in the Standard

Event Status Register (SESR). For a more detailed discussion of error reporting, please refer to the Status and Events section.

**Examples** SYNC:TIMEOUT 10 adjusts the synchronization timeout to a duration of 10 seconds.

SYNC:TIMEOUT? might return 3, indicating the default timeout is 3 seconds.

## SYSTEM:PROPERTIES:CH<x>:BANDwidth? (Query Only)

This query only command returns a list of available bandwidth selections or, if applicable to the sampling module channel, its characteristic bandwidth.

**Group** System

**Syntax** SYSTEM:PROPERTIES:CH<x>:BANDwidth?

**Related Commands** [CH<x>:BANDwidth:LIST?](#), [CH<x>:BANDwidth:VALue](#)

**Examples** SYSTEM:PROPERTIES:CH1:BANDWIDTH? might return SYSTEM:PROPERTIES:CH1:BANDWIDTH 2.0000000000E+010,1.24999997440E+010, indicating that the available bandwidths for Channel 1 are 20 GHz and 12.5 GHz.

SYSTEM:PROPERTIES:CH5:BANDWIDTH? might return SYSTEM:PROPERTIES:CH5:BANDWIDTH 5.0000000000E+010, indicating that the characteristic bandwidth of the sampling module is 50 GHz.

## SYSTEM:PROPERTIES:CH<x>:CAPacitance? (Query Only)

This query only command returns the characteristic load capacitance (if applicable) of the channel specified by x, which can be 1 through 8.

**Group** System

**Syntax** SYSTEM:PROPERTIES:CH<x>:CAPacitance?

**Examples** SYSTEM:PROPERTIES:CH3:CAPACITANCE? might return SYSTEM:PROPERTIES:CH3:CAPACITANCE 4.7500000000E-013, indicating that the characteristic load capacitance for Channel 3 is 0.475 pF.

## SYSTem:PROPERTIES:CH<x>:CLKRec? (Query Only)

This query only command returns a list of the available clock recovery selections available to the trigger system from the channel specified by x, which can be 1 through 8.

**Group** System

**Syntax** SYSTem:PROPERTIES:CH<x>:CLKRec?

**Related Commands** [TRIGGER:CH<x>:CLKRec:LIST?](#), [TRIGGER:CH<x>:CLKRec:VALue](#), [TRIGGER:CLKRec:SOUrce](#)

**Examples** SYSTem:PROPERTIES:CH1:CLKREC? might return  
SYSTem:PROPERTIES:CH1:CLKREC NONE,OC12,OC48.

## SYSTem:PROPERTIES:CH<x>:CLKUser? (No Query Form)

This query returns the allowable range for a User (custom) clock rate for Ch<x> as <nr3> - <nr3>. The channel specified must be from an optical sampling module that provides continuous clock-recovery capability in order to return a range; if not, "NONE" is returned.

The allowable range is the range supported by the module providing the specified channel; values within the range returned by this query can be used to specify the User clock rate when using the TRIGGER:CH<x>:CLKRec:USER command.

**Group** System

**Syntax** SYSTem:PROPERTIES:CH<x>:CLKUser?

**Related Commands** [TRIGGER:CH<x>:CLKRec:RANge](#), [TRIGGER:CH<x>:CLKRec:USER](#)

**Examples** SYSTem:PROPERTIES:CH1:CLKUSER? might return 9.8E9 - 12.6E9, indicating that any User rate between 9.8 Gb/s and 12.6 Gb/s can be defined for Channel 1.

## SYSTem:PROPERTIES:CH<x>:DYNAMIC? (Query Only)

This query only command returns the lower (minimum) and upper (maximum) input signal dynamic limits of the specified channel of the sampling module. The channel is specified by x, which can be 1 through 8.

**Group** System

**Syntax** SYSTem:PROPERTIES:CH<x>:DYNAMIC?

**Examples** SYSTem:PROPERTIES:CH5:DYNAMIC? might return SYSTem:PROPERTIES:CH5:DYNAMIC -5.0000000000E-001, 5.0000000000E-001, indicating that the input dynamic range of Channel 5 is between -0.5 to +0.5, for a total dynamic range of 1.0 (with the applicable unit of measurement determined by the type of sampling module for the channel).

## SYSTem:PROPERTIES:CH<x>:EXTENDER? (Query Only)

This query only command returns the extender cable type that is attached to the channel specified by x, which can be 1 through 8. The extender attaches between the sampling module and the mainframe. This query will return one of the following: NONE, ONEMETER, or TWOMETER, indicating the presence of no extender, a 1 meter extender cable, or a 2 meter extender cable attached to the specified channel.

**Group** System

**Syntax** SYSTem:PROPERTIES:CH<x>:EXTENDER?

**Examples** SYSTem:PROPERTIES:CH8:EXTENDER? might return SYSTem:PROPERTIES:CH8:EXTENDER ONEMETER, indicating that a 1-meter extender cable is present on Channel 8.

## SYSTem:PROPERTIES:CH<x>:FILTER? (Query Only)

This query only command returns a list of the available filter selections for the channel specified by x, which can be 1 through 8.

**Group** System

**Syntax**     `SYSTEM:PROPERTIES:CH<x>:FILTER?`

**Related Commands**     `CH<x>:FILTER:LIST?`, `CH<x>:FILTER:VALUE`

**Examples**     `SYSTEM:PROPERTIES:CH1:FILTER?` might return  
`SYSTEM:PROPERTIES:CH1:FILTER NONE,OC12,OC48,OC192.`

## **SYSTEM:PROPERTIES:CH<x>:IMPEDANCE? (Query Only)**

This query only command returns the characteristic input impedance of the sampling module (if applicable) for the channel specified by x, which can be 1 through 8.

**Group**     System

**Syntax**     `SYSTEM:PROPERTIES:CH<x>:IMPEDANCE?`

**Examples**     `SYSTEM:PROPERTIES:CH3:IMPEDANCE?` might return  
`SYSTEM:PROPERTIES:CH3:IMPEDANCE 5.0000000000E+001,`  
indicating that the characteristic input impedance is 50  $\Omega$  for Channel 3.

## **SYSTEM:PROPERTIES:CH<x>:MODELNUM? (Query Only)**

This query only command returns the sampling module model number of the specified channel.

**Group**     System

**Syntax**     `SYSTEM:PROPERTIES:CH<x>:MODELNUM?`

**Examples**     `SYSTEM:PROPERTIES:CH4:MODELNUM?` might return  
`SYSTEM:PROPERTIES:CH4:MODELNUM "80E04".`

## **SYSTEM:PROPERTIES:CH<x>:NONDESTRUCT? (Query Only)**

This query only command returns the lower (minimum) and upper (maximum) absolute input signal nondestructive limits of the channel specified by x, which can be 1 through 8.



**Group** System

**Syntax** `SYSTEM:PROPERTIES:CH<x>:NONDESTRUCT?`

**Examples** `SYSTEM:PROPERTIES:CH3:NONDESTRUCT?` might return `SYSTEM:PROPERTIES:CH3 NONDESTRUCT -3.000000000000E+000, 3.000000000000E+000`, indicating that the absolute input nondestruct range for Channel 3 is from -3.0 to +3.0 (with the applicable unit of measure determined by the type of sampling module for the channel).

## **SYSTEM:PROPERTIES:CH<x>:OPERATING? (Query Only)**

This query only command returns the lower (minimum) and upper (maximum) absolute input signal operating limits of the sampling module channel specified by x, which can be 1 through 8.

**Group** System

**Syntax** `SYSTEM:PROPERTIES:CH<x>:OPERATING?`

**Examples** `SYSTEM:PROPERTIES:CH5:OPERATING?` might return `SYSTEM:PROPERTIES:CH5:OPERATING -1.600000000000E+000, 1.600000000000E+000`, indicating that the absolute input operating range for CH5 is from -1.6 to +1.6 (with the applicable unit of measure determined by the type of sampling module for the channel).

## **SYSTEM:PROPERTIES:CH<x>:PRANGE? (Query Only)**

Like the `PHASeref:CH<x>:RANGE?` query, this query returns the guaranteed frequency range of the Phase Reference Module corresponding to CH<x>, but it returns the range as part of the system-properties query results. See the [PHASeref:CH<x>:RANGE?](#) query for more information.

**Group** System

**Syntax** `SYSTEM:PROPERTIES:CH<x>:PRANGE?`

**Related Commands** [PHASeref:CH<x>:RANGE?](#), [PHASeref:CH<x>:STATUS?](#)

**Examples**     `SYSTEM:PROPERTIES:CH2:PRANGE?` might return `8.000GHZ-60.00GHZ` for an 82A04 module installed in the C1C2 small-module compartment.

## **SYSTEM:PROPERTIES:CH<x>:PROBE:DYNAMIC? (Query Only)**

This query only command returns the lower (minimum) and upper (maximum) output signal dynamic limits of a probe attached to the channel specified by x, which can be 1 through 8. The actual input dynamic range of the probe (and channel) can be calculated by dividing the dynamic range limits of the probe by the scale factor of the probe.

**Group**     System

**Syntax**     `SYSTEM:PROPERTIES:CH<x>:PROBE:DYNAMIC?`

**Related Commands**     [SYSTEM:PROPERTIES:CH<x>:PROBE:SCALE?](#)

**Examples**     `SYSTEM:PROPERTIES:CH3:PROBE:DYNAMIC?` might return `SYSTEM:PROPERTIES:CH3:PROBE DYNAMIC -4.0000000000E-001,4.0000000000E-001`, indicating that the output dynamic range of the probe attached to Channel 3 is -4.0 to +4.0 (with the applicable unit of measure determined by the type of probe and sampling module for the channel).

## **SYSTEM:PROPERTIES:CH<x>:PROBE:IMPEDANCE? (Query Only)**

This query only command returns the characteristic input impedance of a probe attached to the channel specified by x, which can be 1 through 8.

**Group**     System

**Syntax**     `SYSTEM:PROPERTIES:CH<x>:PROBE:IMPEDANCE?`

**Examples**     `SYSTEM:PROPERTIES:CH1:PROBE:IMPEDANCE?` might return `SYSTEM:PROPERTIES:CH1:PROBE IMPEDANCE 1.0000000000E+005`, indicating that the characteristic input impedance of the probe attached to Channel 1 is 100 kΩ.

## SYSTem:PROPERTIES:CH<x>:PRObe:MODEInum? (Query Only)

This query only command returns the model number of the probe attached to the channel specified by x, which can be 1 through 8.

**Group** System

**Syntax** SYSTem:PROPERTIES:CH<x>:PRObe:MODEInum?

**Examples** SYSTem:PROPERTIES:CH3:PROBE:MODELNUM? might return  
SYSTem:PROPERTIES:CH3:PROBE MODELNUM "P6209", indicating that a Tektronix P6209 FET probe is attached to Channel 3.

## SYSTem:PROPERTIES:CH<x>:PRObe:SCALE? (Query Only)

This query only command returns the scale factor of a probe attached to the channel specified by x, which can be 1 through 8.

**Group** System

**Syntax** SYSTem:PROPERTIES:CH<x>:PRObe:SCALE?

**Examples** SYSTem:PROPERTIES:CH3:PROBE:SCALE? might return  
SYSTem:PROPERTIES:CH3:PROBE SCALE 1.000000000000E-001,  
indicating that the probe attached to Channel 3 has a probe characteristic scale factor of 0.1 (that is, a 10X attenuation from input to output).

## SYSTem:PROPERTIES:CH<x>:PRObe:SERIALnum? (Query Only)

This query only command returns the serial number of the probe attached to the specified channel.

**Group** System

**Syntax** SYSTem:PROPERTIES:CH<x>:PRObe:SERIALnum?

**Examples** SYSTem:PROPERTIES:CH3:PROBE:SERIALNUM? might return  
SYSTem:PROPERTIES:CH3:PROBE:SERIALNUM "B012288".

## SYSTem:PROPERTIES:CH<x>:RISetime? (Query Only)

This query only command returns the characteristic risetime of the sampling module (if applicable) for the channel specified by x, which can be 1 through 8.

**Group** System

**Syntax** SYSTem:PROPERTIES:CH<x>:RISetime?

**Examples** SYSTem:PROPERTIES:CH4:RISETIME? might return  
 SYSTem:PROPERTIES:CH4:RISETIME 1.74999997132E-011, indicating that the characteristic risetime for Channel 4 is 17.5 ps.

## SYSTem:PROPERTIES:CH<x>:SERialnum? (Query Only)

This query only command returns the module serial number of the channel specified by x, which can be 1 through 8.

**Group** System

**Syntax** SYSTem:PROPERTIES:CH<x>:SERialnum?

**Examples** SYSTem:PROPERTIES:CH4:SERIALNUM? might return  
 SYSTem:PROPERTIES:CH4:SERIALNUM "B013456" as the serial number of the module occupying channel 4.

## SYSTem:PROPERTIES:CH<x>:TEKPDriver? (Query Only)

This query only command returns the TEKPROBE driver revision number for the channel specified by x, which can be 1 through 8.

**Group** System

**Syntax** SYSTem:PROPERTIES:CH<x>:TEKPDriver?

**Examples** SYSTem:PROPERTIES:CH4:TEKPDIVER? might return  
 SYSTem:PROPERTIES:CH4:TEKPDIVER "1.00", indicating the TEKPROBE driver revision number for Channel 4.

## SYSTem:PROPERTIES:CH<x>:TEKVersion? (Query Only)

This query only command returns the TEKPROBE version number for the channel specified by x, which can be 1 through 8.

**Group** System

**Syntax** SYSTem:PROPERTIES:CH<x>:TEKVersion?

**Examples** SYSTem:PROPERTIES:CH4:TEKVERSION? might return  
SYSTem:PROPERTIES:CH4:TEKVERSION "3.00", indicating the  
TEKPROBE version number for Channel 4.

## SYSTem:PROPERTIES:CH<x>:WLENGth? (Query Only)

This query only command returns a list of the available factory and user wavelength selections for the channel specified by x, which can be 1 through 8.

**Group** System

**Syntax** SYSTem:PROPERTIES:CH<x>:WLENGth?

**Related Commands** [CH<x>:WLENGth:LIST?](#), [CH<x>:WLENGth:VALue](#)

**Examples** SYSTem:PROPERTIES:CH2:WLENGTH? might return  
SYSTem:PROPERTIES:CH2:WLENGTH FACTORY,1550,1310,USER,1000.

## SYSTem:PROPERTIES:EFEHWver? (Query Only)

This query only command returns the version number of the Electrical Front End (EFE) circuit board.

**Group** System

**Syntax** SYSTem:PROPERTIES:EFEHWver?

**Examples** SYSTem:PROPERTIES:EFEHWVER? might return  
SYSTem:PROPERTIES:EFEHWVER? 1.

## SYSTem:PROPERTIES:EFESERialnum? (Query Only)

This query only command returns the serial number of the Electrical Front End (EFE) circuit board.

**Group** System

**Syntax** SYSTem:PROPERTIES:EFESERialnum?

**Examples** SYSTem:PROPERTIES:EFESERIALNUM? might return SYSTem:PROPERTIES:EFESERIALNUM? "05G7T8".

## SYSTem:PROPERTIES:GROup<x>:TEKPVersion? (Query Only)

This query only command returns the version number of the TEKPROBE controller hardware specified by x, which can be 1 or 2.

**Group** System

**Syntax** SYSTem:PROPERTIES:GROup<x>:TEKPVersion?

**Examples** SYSTem:PROPERTIES:GROUP1:TEKPVERSION? might return SYSTem:PROPERTIES:GROUP1:TEKPVERSION "1331-00", indicating the version of the TEKPROBE controller for Group 1.

## SYSTem:PROPERTIES:MAInframe:KEY? (Query Only)

This query only command returns the key used to enable instrument options.

**Group** System

**Syntax** SYSTem:PROPERTIES:MAInframe:KEY?

**Examples** SYSTem:PROPERTIES:MAINFRAME:KEY? might return SYSTem:PROPERTIES:MAINFRAME KEY "WEMYMKKKKMYYYYYBBBBBAAAAA".

## SYSTEM:PROPERTIES:MAINFRAME:MODELNUM? (Query Only)

This query only command returns the model number of the mainframe instrument.

**Group** System

**Syntax** SYSTEM:PROPERTIES:MAINFRAME:MODELNUM?

**Examples** SYSTEM:PROPERTIES:MAINFRAME:MODELNUM? might return  
SYSTEM:PROPERTIES:MAINFRAME MODELNUM "DSA8300".

## SYSTEM:PROPERTIES:MAINFRAME:SERIALNUM? (Query Only)

This query only command returns the serial number of the mainframe instrument.

**Group** System

**Syntax** SYSTEM:PROPERTIES:MAINFRAME:SERIALNUM?

**Examples** SYSTEM:PROPERTIES:MAINFRAME:SERIALNUM? might return  
SYSTEM:PROPERTIES:MAINFRAME:SERIALNUM "B010247".

## SYSTEM:PROPERTIES:MAINFRAME:SWVERSION? (Query Only)

This query only command returns the software version number of the mainframe instrument.

**Group** System

**Syntax** SYSTEM:PROPERTIES:MAINFRAME:SWVERSION?

**Examples** SYSTEM:PROPERTIES:MAINFRAME:SWVERSION? might return  
SYSTEM:PROPERTIES:MAINFRAME:SWVERSION "6.0.0.0".

## SYSTEM:PROPERTIES:MAINFRAME:UNIQUEID? (Query Only)

This query only command returns the id number of the instrument. The id number is used for instrument options.

**Group** System

**Syntax** `SYSTEM:PROPERTIES:MAINFRAME:UNIQUEID?`

**Examples** `SYSTEM:PROPERTIES:MAINFRAME:UNIQUEID?` might return  
`SYSTEM:PROPERTIES:MAINFRAME UNIQUEID "00:D0:C9:B7:55:F0"`.

## **SYSTEM:PROPERTIES:OFEHWver? (Query Only)**

This query only command returns the Optical Front-End (OFE) board version.

**Group** System

**Syntax** `SYSTEM:PROPERTIES:OFEHWver?`

**Examples** `SYSTEM:PROPERTIES:OFEHWVER?` might return  
`SYSTEM:PROPERTIES:OFEHWVER 1`, showing that the OFE board version is 1.

## **SYSTEM:PROPERTIES:OFESERIALnum? (Query Only)**

This query only command returns the serial number of the Optical Front End (OFE) circuit board.

**Group** System

**Syntax** `SYSTEM:PROPERTIES:OFESERIALnum?`

**Examples** `SYSTEM:PROPERTIES:OFESERIALNUM?` might return `SYSTEM:PROPERTIES:OFESERIALNUM? "05G7T8"`.

## **SYSTEM:PROPERTIES:ONTime? (Query Only)**

This query only command returns the on time (in hours) for the current power-on cycle of the instrument.

**Group** System



**Syntax**    `SYSTem:PROPErties:ONTime?`

**Examples**    `SYSTEM:PROPERTIES:ONTIME?` might return `SYSTEM:PROPERTIES:ONTIME 1.50`, indicating the on time for the instrument for the current power-on cycle is 1.5 hours.

## **SYSTem:PROPErties:POWErups? (Query Only)**

This query only command returns the total number of times that the instrument has been powered on.

**Group**    System

**Syntax**    `SYSTem:PROPErties:POWErups?`

**Examples**    `SYSTEM:PROPERTIES:POWERUPS?` might return `SYSTEM:PROPERTIES:POWERUPS 8`, indicating that the instrument has been powered on a total of eight times.

## **SYSTem:PROPErties:PROCHWver? (Query Only)**

This query only command returns the hardware version of the Processor circuit board in the instrument.

**Group**    System

**Syntax**    `SYSTem:PROPErties:PROCHWver?`

**Examples**    `SYSTEM:PROPERTIES:PROCHWVER?` might return `SYSTEM:PROPERTIES:PROCHWVER 1`.

## **SYSTem:PROPErties:TBHWver? (Query Only)**

This query only command returns the version number of the Timebase (TB) circuit board in the instrument.

**Group**    System

**Syntax**    `SYSTEM:PROPERTIES:TBHWVER?`

**Examples**    `SYSTEM:PROPERTIES:TBHWVER?` might return  
`SYSTEM:PROPERTIES:TBHWVER 1.`

## **SYSTEM:PROPERTIES:TBSerialnum? (Query Only)**

This query only command returns the serial number of the Timebase (TB) circuit board.

**Group**    System

**Syntax**    `SYSTEM:PROPERTIES:TBSerialnum?`

**Examples**    `SYSTEM:PROPERTIES:TBSerialnum?` might return `SYSTEM:PROPERTIES:TBSerialnum? "05G7T8"`.

## **SYSTEM:PROPERTIES:TOTALontime? (Query Only)**

This query only command returns the total accumulated power-on time (in hours) across all power cycles.

**Group**    System

**Syntax**    `SYSTEM:PROPERTIES:TOTALontime?`

**Examples**    `SYSTEM:PROPERTIES:TOTALontime?` might return  
`SYSTEM:PROPERTIES:TOTALontime 533.47`, indicating the total power-on time for the instrument is 533.47 hours.

## **TDR? (Query Only)**

This query only command returns all of the TDR parameters.

**Group**    TDR

**Syntax**    `TDR?`

**Examples** TDR? might return TDR:CH1:UNITS RHO; STEP:STATE 0; DESKEW 0.0000000000; POLARITY PLUS; :TDR:CH2:UNITS RHO; STEP:STATE 0; DESKEW 0.0000000000; POLARITY PLUS; :TDR:CH3:UNITS RHO; STEP:STATE 0;DESKEW 10.5000000000;POLARITY MINUS; :TDR:CH4:UNITS OHM; STEP:STATE 1;DESKEW -11.0000000000;POLARITY MINUS; :TDR:CH5:UNITS RHO; STEP:STATE 0; DESKEW 0.0000000000; POLARITY PLUS; :TDR:CH6:UNITS RHO; STEP:STATE 0; DESKEW 0.0000000000; POLARITY PLUS; :TDR:CH7:UNITS RHO; STEP:STATE 0;DESKEW 0.0000000000; POLARITY PLUS; :TDR:CH8:UNITS RHO; STEP:STATE 0; DESKEW 0.0000000000; POLARITY PLUS; :TDR:INTRATE 100.

## TDR:CH<x>:DIFfpreset (No Query Form)

This command (no query form) sets the instrument to predefined differential TDR settings for the channel pair specified by x, which can be 1 through 8. A channel pair are the channels assigned to each small compartment. You can specify either channel of a channel pair.

A TDR differential preset on a channel pair makes the following changes:

- Turns on the acquisition and display for the selected channel pair.
- Turns on the TDR step for the selected channel pair (odd channel step polarity set to positive, even channel step polarity set to negative).
- Sets the vertical scale, offset, and position for the selected channel pair such that the vertical region corresponding to the front-panel connector is placed approximately at center screen (vertically).
- Sets the horizontal scale and position such that the incident TDR step is on screen.
- Sets the TDR:REF10MHz mode to lock to Internal 10MHz.
- Sets the trigger source to TDR. The Internal Clock Rate is not changed.

---

**NOTE.** *If you are using a sampling module extender cable (or multiple extender cables of differing lengths), it is not possible to view incident TDR steps on those channels not attached to the longest cable due to the effect of extender cables on the minimum horizontal position.*

---

**Group** TDR

**Syntax** TDR:CH<x>:DIFfpreset

**Related Commands** [TDR:CH<x>:UNIts](#), [TDR:CH<x>:STEP:POLarity](#), [TDR:CH<x>:STEP:STATE](#), [TDR:CH<x>:PRESET](#)

**Examples** TDR:CH3:DIFFPRESET performs a TDR preset on Channels 3 and 4.

## TDR:CH<x>:PRESET (No Query Form)

This command (no query form) sets the instrument to predefined TDR settings for the channel specified by x, which can be 1 through 8.

A TDR preset on any channel makes the following changes:

- Turns on the acquisition and display for the selected channel.
- Turns on the TDR step for the selected channel (of the selected step polarity).
- Sets the vertical scale, offset, and position for the selected channel such that the vertical region corresponding to the front-panel connector is placed approximately at center screen (vertically).
- Sets the horizontal scale and position such that the incident TDR step is on screen.
- Sets the TDR:REF10MHz mode to lock to Internal 10MHz.
- Sets the trigger source to TDR. The Internal Clock Rate is not changed.

---

**NOTE.** *If you are using a sampling module extender cable (or multiple extender cables of differing lengths), it is not possible to view incident TDR steps on those channels not attached to the longest cable due to the effect of extender cables on the minimum horizontal position.*

---

**Group** TDR

**Syntax** TDR:CH<x>:PRESET

**Related Commands** [TDR:CH<x>:UNIts](#), [TDR:CH<x>:STEP:POLarity](#), [TDR:CH<x>:STEP:STATE](#)

**Examples** TDR:CH3:PRESET sets the instrument to predefined TDR settings for channels 3 and 4.

## TDR:CH<x>:STEP:DESkew

This command sets or queries the time (deskew) at which the step generator for the even-numbered channel asserts the TDR step relative to the odd-numbered channel of the TDR sampling module. Normally, you do not use deskew with single channels; it is a two-channel adjustment. Channels are selected as pairs to set deskew (for example, Channel 1 and Channel 2, Channel 3 and Channel 4, and so on). You can set the deskew on either channel of the channel pair, and both channels are coerced. The deskew is set as a percentage.

<b>Group</b>	TDR
<b>Syntax</b>	TDR:CH<x>:STEP:DESkew <NR3> TDR:CH<x>:STEP:DESkew?
<b>Arguments</b>	NR3 is the percentage of time at which the step generator for the right channel asserts the TDR step relative to the left channel. The range is -100 to +100% with a resolution of 0.1%.
<b>Examples</b>	TDR:CH3:STEP:DESKEW 1.05+001 sets the deskew between Channel 3 and Channel 4 to 10.5%.  TDR:CH3:STEP:DESKEW? might return TDR:CH3:STEP:DESKEW 30.0000000000, indicating that the deskew for Channel 3 is set to 30%.

## TDR:CH<x>:STEP:POLarity

This command sets or queries the polarity (positive- or negative-going) of the TDR output pulses for the specified channel. The channel is specified by x, which can be 1 through 8. This is equivalent to toggling the polarity icon between a rising edge and a falling edge in the TDR Step section of the TDR Setup dialog box for the specified channel.

<b>Group</b>	TDR
<b>Syntax</b>	TDR:CH<x>:STEP:POLarity {PLUS   MINUS} TDR:CH<x>:STEP:POLarity?
<b>Arguments</b>	<ul style="list-style-type: none"> <li>■ PLUS sets positive-going TDR output pulses.</li> <li>■ MINUS sets negative-going TDR output pulses.</li> </ul>

**Examples** TDR:CH2:STEP:POLARITY PLUS sets positive-going TDR output pulses for Channel 2.

TDR:CH3:STEP:POLARITY? might return TDR:CH3:STEP:POLARITY MINUS, indicating that negative-going TDR output pulses are set for Channel 3.

## TDR:CH<x>:STEP:STATE

This command sets or queries whether the TDR step generator is on or off for the specified channel. The channel is specified by x, which can be 1 through 8. This is equivalent to checking the ON control for the specified channel in the TDR Step section of the TDR Setup dialog box.

---

**NOTE.** *You cannot turn on the step generator for a channel that has a probe attached. If the step generator is on and you then attach a probe to the channel, the instrument will turn the step generator off.*

---

**Group** TDR

**Syntax** TDR:CH<x>:STEP:STATE { ON | OFF | NR1 }  
TDR:CH<x>:STEP:STATE?

- Arguments**
- ON enables the TDR generator for the specified channel.
  - OFF disables the TDR generator for the specified channel.
  - NR1 set to 0 disables the TDR step generator for the specified channel; any other value enables the step generator for the specified channel.

**Examples** TDR:CH6:STEP:STATE OFF turns off the TDR generator for Channel 6.

TDR:CH5:STEP:STATE? might return TDR:CH5:STEP:STATE 1, indicating that the TDR generator for Channel 5 is on.

---

**NOTE.** *A query always returns a 0 or 1.*

---

## TDR:CH<x>:UNIts

This command sets or queries the TDR units for the channel specified by x, which can be 1 through 8. This is equivalent to checking the ON control for the specified channel in the TDR Step section of the TDR Setup dialog box and then selecting the units from the pulldown list for the specified channel. The units of measure commonly used in TDR are units of rho ( $\rho$ ) measured on the vertical axis.

**NOTE.** *These units apply only when the TDR step is on for the specified channel; when the TDR step is on, the vertical units setting for the specified channel is overwritten.*

<b>Group</b>	TDR
<b>Syntax</b>	TDR:CH<x>:UNITS {VOLT   Rho   Ohm} TDR:CH<x>:UNITS?
<b>Related Commands</b>	<a href="#">TDR:CH&lt;x&gt;:PRESET</a> , <a href="#">CH&lt;x&gt;:UNITS</a>
<b>Arguments</b>	<ul style="list-style-type: none"> <li>■ VOLT sets the units for the specified channel to volts.</li> <li>■ Rho sets the units for the specified channel to rho.</li> <li>■ Ohm sets the units for the specified channel to ohms.</li> </ul>
<b>Examples</b>	<p>TDR:CH3:UNITS VOLT sets the units for Channel 3 to volts.</p> <p>TDR:CH3:UNITS? might return TDR:CH3:UNITS OHM, indicating that the units for Channel 3 are ohms.</p>

## TDR:EXT10MHZref:FREQ

This command sets or returns the external 10 MHz reference frequency for TDR measurements. The command allows you to adjust the external 10 MHz reference frequency to ensure that the timebase locks when the Lock to Ext 10 MHz mode is set using the TDR:REF10Mhz command.

<b>Group</b>	TDR
<b>Syntax</b>	TDR:EXT10MHZref:FREQ <NR3> TDR:EXT10MHZref:FREQ?
<b>Related Commands</b>	<a href="#">TDR:REF10Mhz</a>
<b>Arguments</b>	<NR3> is the external 10 MHz frequency setting. The range for this argument is $1.0E7 \pm 0.0005\%$ (9.9995 MHz through 10.0005 MHz). The resolution is 0.1 Hz. The default value is 10 MHz.

- Examples** TDR:CH<x>:EXT10MHZREF:FREQ 9.9995E6 sets the TDR external 10 MHz frequency to 9.9995 MHz.
- TDR:CH<x>:EXT10MHZREF:FREQ? might return TDR:EXT10MHZREF:FREQ 10.0000000000E+006, indicating that the TDR external 10 MHz frequency is set to the default value (10 MHz).

## TDR:INTRate

This command sets or queries the TDR (internal clock) rate. The command sets the same parameter as the TRIGger:INTRate command. The instrument uses the internal clock rate to synchronize the generation of TDR step pulses with the subsequent acquisition. The internal clock in the instrument generates a continuous signal with selectable repetition rates of from 25 to 300 kHz in 1 kHz increments. The actual repetition rate may vary by as much as 4% from the specified value.

Sending this command is equivalent to selecting a value for Internal Clock in the TDR (or Trigger) Setup dialog box.

- Group** TDR
- Syntax** TDR:INTRate <NR3>  
TDR:INTRate?
- Arguments** NR3 sets the internal clock rate to any value in the range of 25 kHz to 300 kHz. The actual TDR repetition rate may vary by up to 4% from the specified value. You can input any NR3 value; the instrument will set the internal clock rate to the closest available setting.
- Examples** TDR:INTRATE 100E3 sets the internal clock rate to 100 kHz.
- TDR:INTRATE? might return TDR:INTRATE 0.200000000000E+006, indicating that the internal clock rate is set to 200 kHz (the default value).

## TDR:REF10Mhz

This command sets or returns the source (internal or external) for the TDR 10 MHz reference frequency used when the trigger source is set to TDR.

- Group** TDR



**Syntax** TDR:REF10Mhz {LOCKExt | LOCKInt}  
TDR:REF10Mhz?

**Related Commands** [TDR:EXT10MHZref:FREQ](#)

**Arguments** LOCKExt locks the TDR timebase to an external 10 MHz reference signal.  
LOCKInt locks the TDR timebase to the internal 10 MHz reference signal.

**Examples** TDR:REF10MHZ LOCKInt sets the instrument to use the internal timebase.  
TDR:REF10MHZ? might return LOCKInt, indicating the instrument is locked to the internal timebase.

## TIME

This command sets or queries the time that the instrument can display.

**Group** Miscellaneous

**Syntax** TIME <QString>  
TIME?

**Related Commands** [DATE](#)

**Arguments** QSTRING is time in the form "hh:mm:ss".

- hh is the hour number from 1 to 24.
- mm is the minute number in the hour from 0 to 59.
- ss is the seconds number in the minute from 0 to 59.

---

**NOTE.** *There must be a colon between the hh and mm, and between the mm and ss.*

---

**Examples** TIME "01:24:00" sets the time to 01:24 AM.  
TIME? might return TIME 14:30:00, indicating that the time is 2:30 PM.

## TRIGger? (Query Only)

This query only command returns all the trigger parameters for the instrument.

<b>Group</b>	Trigger
<b>Syntax</b>	TRIGger?
<b>Examples</b>	TRIGGER? might return TRIG:GATE 0;HIF 0;HOLD 5.000000000000E-6;INTR 200.000000000000E+3;LEV 0.0000;MOD NORM;SLO RIS;SOU CLKREC;STATE TRIGGERED;CLKR:SOU CH1;:TRIG:CH1:CLKR:VAL USE;USE 9.800000000000E+9;:TRIG:CH2:CLKR:VAL NON;USE 0.0000.

## TRIGger:AUTOSync (No Query Form)

The auto (pattern) sync operation attempts to automatically determine the data rate, clock to data rate ratio, and the pattern length of a repeating pattern. It does this by examining the data stream and the clock associated with that data stream and varying these parameters until a stable bit pattern is displayed. Auto sync can be setup to attempt to detect any or all of these parameters.

Invoking this command executes all Auto sync operations.

Any pattern sync trigger parameter that has its AutoSync option enabled will synchronize with the currently selected waveform.

The command supports both the \*OPC and \*OPC? Operation Complete capabilities.

<b>Group</b>	Trigger
<b>Syntax</b>	TRIGger:AUTOSync {EXECute}
<b>Arguments</b>	EXECute executes the AutoSync operations.
<b>Examples</b>	TRIGGER:AUTOSYNC EXECUTE executes all the AutoSync operations.

## TRIGger:AUTOSync:DATARate

This command sets the AutoSync data rate selection option to ON or OFF, so that when AutoSync executes it will or will not attempt to determine the data rate of the currently selected waveform.

The query form returns the status of the option.

<b>Group</b>	Trigger
<b>Syntax</b>	TRIGger:AUTOSync:DATARate { ON   OFF   NR1 } TRIGger:AUTOSync:DATARate?
<b>Arguments</b>	<ul style="list-style-type: none"> <li>■ ON enables the AutoSync data rate function.</li> <li>■ OFF disables the AutoSync data rate function.</li> <li>■ NR1 set to 0 disables the AutoSync data rate function; any other value enables the AutoSync data rate function.</li> </ul>
<b>Examples</b>	<p>TRIGGER:AUTOSYNC:DATARATE ON sets the AutoSync data rate selection option to ON, so that AutoSync attempts to determine the data rate of the selected waveform when it executes.</p> <p>TRIGGER:AUTOSYNC:DATARATE? might return TRIGGER:AUTOSYNC:DATARATE 0, indicating that AutoSync is not set to determine the data rate when it executes.</p>

## TRIGger:AUTOSync:DCRAtio

This command sets the AutoSync data-to-clock ratio selection option to ON or OFF, so that when AutoSync executes it will or will not attempt to determine the data-to-clock ratio for the currently selected waveform.

The query form returns the status of the option.

<b>Group</b>	Trigger
<b>Syntax</b>	TRIGger:AUTOSync:DCRAtio { ON   OFF   NR1 } TRIGger:AUTOSync:DCRAtio?
<b>Arguments</b>	<ul style="list-style-type: none"> <li>■ ON enables the AutoSync data-to-clock ratio function.</li> <li>■ OFF disables the AutoSync data-to-clock ratio function.</li> <li>■ NR1 set to 0 disables the AutoSync data-to-clock ratio function; any other value enables the AutoSync data-to-clock ratio function.</li> </ul>
<b>Examples</b>	<p>TRIGGER:AUTOSYNC:DCRATIO ON sets the AutoSync data-to-clock ratio selection option to ON, so that AutoSync attempts to determine the data-to-clock ratio of the selected waveform when it executes.</p>

TRIGGER:AUTOSYNC:DCRATIO? might return TRIGGER:AUTOSYNC:DCRATIO 0, indicating that AutoSync is not set to attempt to determine the data-to-clock ratio setting when it executes.

## TRIGger:AUTOSync:PLENgtH

This command sets the AutoSync pattern length selection option to ON or OFF, so that when AutoSync executes it will or will not attempt to determine the pattern length of the currently selected waveform.

The query form returns the status of the option.

**Group** Trigger

**Syntax** TRIGger:AUTOSync:PLENgtH { ON | OFF | NR1 }  
TRIGger:AUTOSync:PLENgtH?

- Arguments**
- ON enables the AutoSync pattern length function.
  - OFF disables the AutoSync pattern length function.
  - NR1 set to 0 disables the AutoSync pattern length function; any other value enables the AutoSync pattern length function.

**Examples**

TRIGGER:AUTOSYNC:PLENGTH ON sets the AutoSync pattern length selection option to ON, so that AutoSync attempts to determine the pattern length of the selected waveform when it executes.

TRIGGER:AUTOSYNC:PLENGTH? might return TRIGGER:AUTOSYNC:PLENGTH 0, indicating that AutoSync is not set to automatically determine the pattern length of the selected waveform when it executes.

## TRIGger:CH<x>:CLKRec:LIST? (Query Only)

This query returns a list of the clock recovery selections that are available for the channel specified by x, which can be 1 through 4. The available choices are dependent on the optical sampling or accessory module that is installed.

The optical modules have differing valid settings. Use the <>:LIST? query to determine the valid {Clock Recovery | Filter | Bandwidth | Wavelength} settings for your optical module. Features of optical modules are listed in their respective user manuals. Features of currently installed optical modules can be viewed using the System Properties dialog.

<b>Group</b>	Trigger
<b>Syntax</b>	TRIGger:CH<x>:CLKRec:LIST?
<b>Examples</b>	TRIGGER:CH1:CLKREC:LIST? might return TRIGGER:CH1:CLKREC:LIST NONE,OC12,OC48.

## TRIGger:CH<x>:CLKRec:RANge

Sets or queries the allowable range for a User (custom) clock rate as <NR3>-<NR3> for the channel specified by X, which can be 1 through 4. (The available choices for x are dependent on the sampling module that is installed.) The channel specified must be from an sampling module that provides continuous clock-recovery capability in order to return a range; otherwise, "NONE" is returned.

The allowable range is the range supported by the module providing the specified channel; values within the range returned by this query can be used to specify the User clock rate when using the TRIGger:CH<x>:CLKRec:USER command.

<b>Group</b>	Trigger
<b>Syntax</b>	TRIGger:CH<x>:CLKRec:RANge TRIGger:CH<x>:CLKRec:RANge?
<b>Related Commands</b>	<a href="#">TRIGger:CH&lt;x&gt;:CLKRec:USER</a> , <a href="#">SYSTEM:PROPERTIES:CH&lt;x&gt;:CLKUser?</a>
<b>Examples</b>	TRIGGER:CH1:CLKREC:RANGE? might return 9.8E9 - 12.6E9, indicating that any user rate between 9.8 Gb/s and 12.6 Gb/s can be defined for Channel 1.  TRIGGER:CH2:CLKREC:RANGE? might return NONE, indicating that continuous clock recovery is not supported by the sampling module that contains Channel 2; therefore, no custom user rate can be defined for Channel 2.

## TRIGger:CH<x>:CLKRec:USER (No Query Form)

This command defines the User (custom) clock rate (<nr3>) for the channel specified (CH<x>), which can be 1 through 4. (The available choices are dependent on the optical sampling module that is installed.) The channel specified must be from an optical sampling module that provides continuous clock-recovery capability. The clock rate defined must be in the range available for the channel/module chosen.

The query form returns the user-defined rate in effect. The default rate is the minimum value of the adjustable range.

**Group** Trigger

**Syntax** TRIGGER:CH<x>:CLKRec:USER <nr3>

**Related Commands** [TRIGGER:CH<x>:CLKRec:RANGE](#), [SYSTEM:PROPERTIES:CH<x>:CLKUser?](#), [SYSTEM:PROPERTIES:CH<x>:CLKRec?](#)

**Examples** TRIGGER:CH1:CLKREC:USER 10.52E9 defines the user-specified clock recovery rate as 10.52 Gb/s.

TRIGGER:CH1:CLKREC:USER? might return 9.8E9, indicating the current user-defined rate is 9.8 Gb/s.

## TRIGGER:CH<x>:CLKRec:VALue

This command sets or queries the clock recovery value that is applied to the channel specified by x, which can be 1 through 4. The clock-recovery selections require hardware support that may not be provided by your sampling module; see your module user manual to determine which clock-recovery selections your module supports. You can also use the [SYSTEM:PROPERTIES:CH<x>:CLKRec?](#) query to determine clock recovery options available for the specified channel.

If you select a clock recovery setting that your sampling module does not support, you will receive Execution Error code 220 with error message "Parameter Error." (See [Messages](#) for general information on programming interface messages.)

**Group** Trigger

**Syntax** TRIGGER:CH<x>:CLKRec:VALue { NONE | ENET1250 | ENET2500 | ENET3125 | ENET9953 | ENET10313 | ENET11096 | FC1063 | FC2125 | FC3188 | FC4250 | FC10519 | FEC2666 | FEC10664 | FEC10709 | INFINIBAND | INPUT\_DATA | OC1 | OC3 | OC9 | OC12 | OC18 | OC24 | OC36 | OC48 | OC96 | OC192 | PCIX2500 | SATA1500 | SATA3000 | SATA6000 | SFEC12500 | USER | VSR5\_3318 | XAUI3125 | XAUI6250 }  
TRIGGER:CH<x>:CLKRec:VALue?

**Related Commands** [TRIGGER:CH<x>:CLKRec:LIST?](#), [TRIGGER:CLKRec:SOURce](#)

- Arguments**
- NONE disables clock recovery for the specified channel.
  - ENET1250 selects clock recovery for the 1.250 Gb/s bit rate, in accordance with the IEEE 802.3 standard.
  - ENET2500 selects clock recovery for the 2.500 Gb/s bit rate (2x Gigabit Ethernet).
  - ENET3125 selects clock recovery for the 3.125 Gb/s bit rate, in accordance with the IEEE 802.3ae-2002 standard.
  - ENET9953 selects clock recovery for the 9.9533 Gb/s bit rate, in accordance with the IEEE 802.3ae-2002 standard.
  - ENET10313 selects clock recovery for the 10.3125 Gb/s bit rate, in accordance with the IEEE 802.3ae-2002 standard.
  - ENET11096 selects clock recovery for the 11.096 Gb/s bit rate, which is a forward error-correction rate for 10GbE (10.3125 Gb/s per IEEE 802.3ae-2002).
  - FC1063 selects clock recovery for the 1.0625 Gb/s bit rate, in accordance with the ANSI X3.230-1994 standard.
  - FC2125 selects clock recovery for the 2.125 Gb/s bit rate, in accordance with the ANSI Fibre Channel Physical Interface (FC-PI) Rev 11 draft standard.
  - FC3188 selects clock recovery for the 3.1875 Gb/s bit rate, in accordance with the proposed 10 Gigabit Fibre Channel (10GFC) standard.
  - FC4250 selects clock recovery for the 4.250 Gb/s bit rate, in accordance with the ANSI Fibre Channel Physical Interface (FC-PI) Rev 11 draft standard.
  - FC10519 selects clock recovery for the 10.518750 Gb/s bit rate, in accordance with the proposed 10 Gigabit Fibre Channel (10GFC) standard.
  - FEC2666 selects clock recovery for the 2.6660571 Gb/s bit rate, in accordance with the ITU-T G.975 Recommendation.
  - FEC10664 selects clock recovery for the 10.6642 Gb/s bit rate, in accordance with the ITU-T G.975 Recommendation.
  - FEC10709 selects clock recovery for the 10.709225 Gb/s bit rate, in accordance with the ITU-T G.709 Draft Recommendation.
  - INFINiBand selects clock recovery for the 2.500 Gb/s bit rate, in accordance with the InfiniBand standard.
  - INPUT\_DATA selects the data inputs of the clock recovery module as the triggering source.
  - OC1 selects clock recovery for the 51.84 Mb/s bit rate, in accordance with the Sonet/SDH OC-1/STM-0 standard.
  - OC3 selects clock recovery for the 155.52 Mb/s bit rate, in accordance with the Sonet/SDH OC-3/STM-1 standard.

- OC9 selects clock recovery for the 466.56 Mb/s bit rate, in accordance with the Sonet/SDH OC-9/STM-3 standard.
- OC12 selects clock recovery for the 622.08 Mb/s bit rate, in accordance with the Sonet/SDH OC-12/STM-4 standard.
- OC18 selects clock recovery for the 933.12 Mb/s bit rate, in accordance with the Sonet/SDH OC-18/STM-6 standard.
- OC24 selects clock recovery for the 1.24416 Gb/s bit rate, in accordance with the Sonet/SDH OC-24/STM-8 standard.
- OC36 selects clock recovery for the 1.86624 Gb/s bit rate, in accordance with the Sonet/SDH OC-36/STM-12 standard.
- OC48 selects clock recovery for the 2.48832 Gb/s bit rate, in accordance with the Sonet/SDH OC-48/STM-16 standard.
- OC96 selects clock recovery for the 4.97664 Gb/s bit rate, in accordance with the Sonet/SDH OC-96/STM-32 standard.
- OC192 selects clock recovery for the 9.95328 Gb/s bit rate, in accordance with the Sonet/SDH OC-192/STM-64 standard.
- PCIX2500 selects clock recovery for the 2.5 Gb/s bit rate, in accordance with the PCI Express Base specification.
- SATA1500 selects clock recovery for the 1.5 Gb/s bit rate, in accordance with the Serial ATA 1.0a specification.
- SATA3000 selects clock recovery for the 3.0 Gb/s bit rate (2nd generation SATA rate).
- SATA6000 selects clock recovery for the 6.0 Gb/s bit rate (potential 3<sup>rd</sup> generation SATA rate).
- SFEC12500 selects clock recovery for the 12.5 Gb/s bit rate (potential ITU-T G.975 related forward error correction rate).
- USER selects clock recovery for the user-specified bit rate, when the channel specified is for an optical sampling module that supports continuous clock recovery (see the `TRIGger:CH<x>:CLKRec:USER` command).
- VSR5\_3318 selects clock recovery for the 3.31776 Gb/s bit rate, in accordance with the OIF Implementation Agreement OIF-VSR5-01.0.
- XAUI3125 selects clock recovery for the 3.125 Gb/s bit rate, in accordance with the IEEE 802.3ae-2002 standard.
- XAUI6250 selects clock recovery for the 6.25 Gb/s bit rate (double the 3.125 Gb/s XAUI rate).

### Examples

`TRIGGER:CH1:CLKREC:VALUE OC12` sets the clock-recovery value for Channel 1 to the Sonet/SDH OC-12/STM-4 standard.



TRIGGER:CH1:CLKREC:VALUE? might return TRIGGER:CH1:CLKREC:VALUE OC48, indicating that the clock-recovery value selected for Channel 1 is the Sonet/SDH OC-48/STM-16 standard (data rate of 2488.3 Mb/s).

## TRIGger:CLKRec:CRC:AUTOSConfigdev

This command sets or queries the setup auto-save feature of the connected clock recovery instrument. If Auto-save is enabled, the clock recovery instrument saves the current configuration in non-volatile memory. At the next power on, the clock recovery instrument is returned to the saved configuration.

**Group** Trigger

**Syntax** TRIGger:CLKRec:CRC:AUTOSConfigdev {ON | OFF | <NR1>}  
TRIGger:CLKRec:CRC:AUTOSConfigdev?

- Arguments**
- ON specifies that Auto-save is enabled.
  - OFF specifies that Auto-save is disabled.
  - <NR1>. 1 enables auto-save, 0 disables auto-save.

**Examples** TRIGGER:CLKREC:CRC:AUTOSCONFIGDEV ON enables the auto-save feature on the clock recovery instrument.

TRIGGER:CLKREC:CRC:AUTOSCONFIGDEV? might return TRIGGER:CLKREC:CRC:AUTOSCONFIGDEV 1, indicating that Auto-save is enabled.

---

**NOTE.** 1 is returned if a clock recovery instrument is not present.

---

## TRIGger:CLKRec:CRC:CAPabilities? (Query Only)

Returns the instrument clock recovery device capabilities.

**Group** Trigger

**Syntax** TRIGger:CLKRec:CRC:CAPabilities?

**Returns** Returns an integer representation of a bit mask that indicates the capabilities of the clock recovery instrument. The bit mask definitions are as follows:

- Bit 0 represents HS (High Sensitivity- no loop-through data) capability.
- Bit 1 indicates DDS capability.
- Bit 2 indicates Input analysis capability.
- Bit 3 indicates loop calibration refinement capability.
- Bit 4 indicates trigger option capability.
- Bit 5 indicates jitter spectrum analysis capability.
- Bit 6 indicates USB diagnostics capability.
- Bit 7 indicates golden PLL mode capability.
- Bit 8 indicates optical receiver (DCRJ) capability.
- Bit 9 indicates PCIe PLL tester option capability.
- Bit 10 indicates jitter scope capability.
- Bit 11 indicates DDR capability, allows data input above 14.3 GHz.

Bits are in LSB to MSB order. For example, HS is bit zero (0000000001), DDS is bit one (0000000010), and so on.

Query response when no clock recovery instrument is present: 0.

**Examples** TRIGGER:CLKREC:CRC:CAPABILITIES? might return 40, indicating that both jitter spectrum analysis and loop calibration refinement are present in the clock recovery instrument.

## TRIGger:CLKRec:CRC:CLOCKAmplitude

This command sets or queries the clock output amplitude (in mV) of the clock recovery instrument.

**Group** Trigger

**Syntax** TRIGger:CLKRec:CRC:CLOCKAmplitude <NR3>  
TRIGger:CLKRec:CRC:CLOCKAmplitude?

**Arguments** NR3 is the clock output amplitude. The range is 250 mV to 900 mV.

**Examples** TRIGGER:CLKREC:CRC:CLOCKAMPLITUDE 250E-3 sets the clock output amplitude to 250 mV.

TRIGGER:CLKREC:CRC:CLOCKAMPLITUDE? might return  
:TRIGGER:CLKREC:CRC:CLOCKAMPLITUDE 0.9000000000,  
indicating that the clock amplitude is set to 900 mV.

---

**NOTE.** *NAN is returned if a clock recovery instrument is not present.*

---

## TRIGger:CLKRec:CRC:CLOCKOutput

This command sets or queries the clock output mode of the connected clock recovery instrument.

**Group** Trigger

**Syntax** TRIGger:CLKRec:CRC:CLOCKOutput {ON | OFF | <NR1>}  
TRIGger:CLKRec:CRC:CLOCKOutput?

- Arguments**
- ON enables the clock output on the clock recovery instrument.
  - OFF disables the clock output on the clock recovery instrument.
  - <NR1>. 1 enables the clock output. 0 disables the clock output.

**Examples** TRIGGER:CLKREC:CRC:CLOCKOUTPUT ON enables the clock output of the clock recovery instrument.

TRIGGER:CLKREC:CRC:CLOCKOUTPUT? might return  
:TRIGGER:CLKREC:CRC:CLOCKOUTPUT 1, indicating that the clock output is enabled.

---

**NOTE.** *0 is returned if a clock recovery instrument is not present.*

---

## TRIGger:CLKRec:CRC:CREATESTandard (No Query Form)

This command saves a new clock recovery standard to the non-volatile memory of the connected clock recovery instrument. Use this command to name the new standard which is based on the current settings for NOMFREQUENCY, LOCKRANGE, LOOPBANDWIDTH, PEAKING, and NOMEDGEDENSITY.

**Group** Trigger

**Syntax** TRIGger:CLKRec:CRC:CREATESTandard <Qstring>

**Related Commands**    [TRIGger:CLKRec:CRC:NOMFrequency](#)  
[TRIGger:CLKRec:CRC:LOCKRange](#)  
[TRIGger:CLKRec:CRC:LOOPBandwidth](#)  
[TRIGger:CLKRec:CRC:PEAKing](#)  
[TRIGger:CLKRec:CRC:NOMEDgedensity](#)  
[TRIGger:CLKRec:CRC:DELETESStandard](#)

**Arguments**    QString is a quoted string defines the file name to save the new standard. The filename is case sensitive and spaces are allowed in the filename. Do not use wild card characters.

**Examples**    TRIGGER:CLKREC:CRC:CREATESTANDARD "USER\_10G" saves a new standard named USER\_10G to the non-volatile memory of the clock recovery instrument.

## TRIGger:CLKRec:CRC:DATARate? (Query Only)

This query only command returns the measured data rate (in Hz) from the connected clock recovery instrument.

**Group**    Trigger

**Syntax**    TRIGger:CLKRec:CRC:DATARate?

**Examples**    TRIGGER:CLKREC:CRC:DATARATE? might return  
:TRIGGER:CLKREC:CRC:DATARATE 12.05000000000E+9, indicating that the measured data rate is 12.05 GHz.

---

**NOTE.** NAN is returned if a clock recovery instrument is not present.

---

## TRIGger:CLKRec:CRC:DELETESStandard (No Query Form)

This command deletes the named standard from the connected clock recovery instrument, including any factory-supplied standard.

If a factory installed standard is deleted by mistake, it can be recreated using the TRIGger:CLKRec:CRC:CREATESTANDARD command.

**Group**    Trigger

**Syntax** TRIGger:CLKRec:CRC:DELETESStandard <Qstring>

**Related Commands** [TRIGger:CLKRec:CRC:CREATESTANDARD](#)

**Arguments** QString is a quoted string defines filename of the standard to delete from the clock recovery instrument. The filename is case sensitive.

**Examples** TRIGGER:CLKREC:CRC:DELETESANDARD "USER\_10G" deletes the standard named USER\_10G from the clock recovery instrument.

## TRIGger:CLKRec:CRC:DEVICENAME

This command sets or queries the device name of the connected clock recovery instrument. The name is saved in non-volatile memory on the clock recovery instrument.

**Group** Trigger

**Syntax** TRIGger:CLKRec:CRC:DEVICENAME <Qstring>

**Arguments** QString is a quoted string that defines a device name for the clock recovery instrument. The device name can be from 1 to 11 characters and is case sensitive.

**Examples** TRIGGER:CLKREC:CRC:DEVICENAME "TEK\_80A07" sets the name of the connected clock recovery instrument to TEK\_80A07.

TRIGGER:CLKREC:CRC:DEVICENAME? might return  
:TRIGGER:CLKREC:CRC:DEVICENAME TEK\_80A07 indicating that the name of the currently connected clock recovery instrument is TEK\_80A07.

---

**NOTE.** *Undefined is returned if a clock recovery instrument is not present.*

---

## TRIGger:CLKRec:CRC:EDGEDensity? (Query Only)

This query only command returns the measured edge density (in percent) from the connected clock recovery instrument.

**Group** Trigger

**Syntax** TRIGger:CLKRec:CRC:EDGEDensity?

**Examples** TRIGGER:CLKREC:CRC:EDGE DENSITY? might return  
:TRIGGER:CLKREC:CRC:EDGE DENSITY 31.11091041565, indicating that  
the edge density is 31.11%.

---

**NOTE.** *NAN is returned if a clock recovery instrument is not present.*

---

## TRIGger:CLKRec:CRC:EDGE DENSITYMode

This command sets or queries the edge density mode setting of the connected clock recovery instrument.

**Group** Trigger

**Syntax** TRIGger:CLKRec:CRC:EDGE DENSITYMode {NOMi na l | ON\_Lock}  
TRIGger:CLKRec:CRC:EDGE DENSITYMode?

**Related Commands** [TRIGger:CLKRec:CRC:NOMEDgedensity](#)

- Arguments**
- **NOMi na l** specifies the clock recovery instrument to use the edge density set by the command TRIGger:CLKRec:CRC:NOMEDgedensity.
  - **ON\_Lock** specifies the clock recovery instrument to use the edge density found when the first lock was acquired.

**Examples** TRIGGER:CLKREC:CRC:EDGE DENSITYMODE ON\_LOCK sets the clock recovery instrument to use the edge density found when the first lock was acquired.

TRIGGER:CLKREC:CRC:EDGE DENSITYMODE? might return  
:TRIGGER:CLKREC:CRC:EDGE DENSITYMODE ON\_Lock, indicating  
that the clock recovery instrument is using the edge density measured when lock was acquired.

---

**NOTE.** *NOMINAL is returned if a clock recovery instrument is not present.*

---

## TRIGger:CLKRec:CRC:EQ ualizer

Sets or queries the phase detection equalization setting for the BERTScope CR286A clock recovery instrument. Phase detection equalization enables clock recovery on high ISI signals.

---

**NOTE.** *This command applies only to the BERTScope CR286A instrument.*

---

<b>Group</b>	Trigger
<b>Syntax</b>	TRIGger:CLKRec:CRC:EQAlizer <NR2> TRIGger:CLKRec:CRC:EQAlizer?
<b>Arguments</b>	NR2 sets the CR286A equalizer value. The range is 0 to 10, where 0 = no equalization, 10 = minimum equalization, and 0.1 = maximum equalization.  Query response when no clock recovery instrument is present: NaN.
<b>Examples</b>	TRIGGER:CLKREC:CRC:EQUALIZER 4.6 sets the CR286A equalizer value to 4.6.  TRIGGER:CLKREC:CRC:EQUALIZER? might return 2.2, indicating that the CR286A equalizer setting is 2.2.

### TRIGger:CLKRec:CRC:EXPrev? (Query Only)

This query only command returns the expansion board revision string of the connected clock recovery instrument.

<b>Group</b>	Trigger
<b>Syntax</b>	TRIGger:CLKRec:CRC:EXPrev?
<b>Examples</b>	TRIGGER:CLKREC:CRC:EXPREV? might return :TRIGGER:CLKREC:CRC:EXPREV xxxx. (xxxx represents the actual board revision).

---

**NOTE.** *undefined is returned if a clock recovery instrument is not present.*

---

### TRIGger:CLKRec:CRC:FPGarev? (Query Only)

This query only command returns the FPGA revision string of the connected clock recovery instrument.

<b>Group</b>	Trigger
--------------	---------

**Syntax** TRIGger:CLKRec:CRC:FPGarev?

**Examples** TRIGGER:CLKREC:CRC:FPGAREV? might return  
:TRIGGER:CLKREC:CRC:FPGAREV 2.8.

---

**NOTE.** *undefined is returned if a clock recovery instrument is not present.*

---

## TRIGger:CLKRec:CRC:HALFRate? (Query Only)

Indicates whether the clock outputs of the clock recovery instrument (CRI) are at the full-rate or half of the input signal's data rate.

**Group** Trigger

**Syntax** TRIGger:CLKRec:CRC:HALFRate?

**Returns** 0 indicates the clock recovery instrument is running at full clock rate.  
1 indicates the clock recovery instrument is running at half clock rate.  
Query response when no CRI present: 0. (*Is this a valid statement?*)

**Examples** TRIGGER:CLKREC:CRC:HALFRATE? might return 1, indicating that the clock recovery instrument is running at half the clock rate.

## TRIGger:CLKRec:CRC:HWRev? (Query Only)

This query only command returns the hardware revision string of the connected clock recovery instrument.

**Group** Trigger

**Syntax** TRIGger:CLKRec:CRC:HWRev?

**Examples** TRIGGER:CLKREC:CRC:HWREV? might return :TRIGGER:CLKREC:CRC:HWREV  
xxxx (xxxx represents the hardware revision string).

---

**NOTE.** *undefined is returned if a clock recovery instrument is not present.*

---



## TRIGger:CLKRec:CRC:LOCKCount? (Query Only)

This query only command returns the number of times the connected clock recovery instrument.

**Group** Trigger

**Syntax** TRIGger:CLKRec:CRC:LOCKCount?

**Examples** TRIGGER:CLKREC:CRC:LOCKCOUNT? might return  
:TRIGGER:CLKREC:CRC:LOCKCOUNT 1 indicating that the clock recovery instrument has locked once.

---

**NOTE.** 0 is returned if a clock recovery instrument is not present.

---

## TRIGger:CLKRec:CRC:LOCKFp (No Query Form)

Locks or unlocks the clock recovery instrument front panel controls.

**Group** Trigger

**Syntax** TRIGger:CLKRec:CRC:LOCKFp { ON | OFF | 1 | 0 }

**Arguments** ON or 1 locks the clock recovery instrument front panel controls.  
OFF or 0 unlocks the clock recovery instrument front panel controls.

**Examples** TRIGGER:CLKREC:CRC:LOCKFP 0 unlocks (enables) the front panel controls.

## TRIGger:CLKRec:CRC:LOCKMode

This command sets or queries the lock mode of the connected clock recovery instrument.

**Group** Trigger

**Syntax** TRIGger:CLKRec:CRC:LOCKMode {MANua] | AUTOMatic}  
TRIGger:CLKRec:CRC:LOCKMode?

<b>Arguments</b>	<ul style="list-style-type: none"> <li>■ <b>MANua1</b> forces the clock recovery instrument to attempt to acquire a signal lock.</li> <li>■ <b>AUTOMATIC</b> sets the clock recovery instrument to automatically re-acquire a signal lock if the phase error exceeds the set limits or the recovered clock is unstable. This is the default setting.</li> </ul>
<b>Examples</b>	<p><b>TRIGGER:CLKREC:CRC:LOCKMODE AUTOMATIC</b> sets the signal lock mode of the clock recovery instrument to automatic.</p> <p><b>TRIGGER:CLKREC:CRC:LOCKMODE?</b> might return  <b>:TRIGGER:CLKREC:CRC:LOCKMODE AUTO</b>, indicating that the lock mode is set to the automatic setting.</p>

---

**NOTE.** *AUTOMATIC is returned if a clock recovery instrument is not present.*

---

## TRIGger:CLKRec:CRC:LOCKRange

This command sets or queries the locking range (in Hz) about the nominal frequency setting of the connected clock recovery instrument.

<b>Group</b>	Trigger
<b>Syntax</b>	<p><b>TRIGger:CLKRec:CRC:LOCKRange &lt;NR3&gt;</b>  <b>TRIGger:CLKRec:CRC:LOCKRange?</b></p>
<b>Arguments</b>	NR3 sets the plus and minus frequency lock. The range is 10 MHz to 500 MHz.
<b>Examples</b>	<p><b>TRIGGER:CLKREC:CRC:LOCKRANGE 10E6</b> sets the range of the signal to deviate plus or minus 10 MHz from the frequency of the selected standard and remained locked.</p> <p><b>TRIGGER:CLKREC:CRC:LOCKRANGE?</b> might return  <b>:TRIGGER:CLKREC:CRC:LOCKRANGE 50.000000000000E6</b>, indicating that the signal frequency can deviate 50 MHz from the frequency standard and remained locked.</p>

---

**NOTE.** *NAN is returned if a clock recovery instrument is not present.*

---

## TRIGger:CLKRec:CRC:LOCKState? (Query Only)

This query only command returns the lock state of the clock recovery instrument.

<b>Group</b>	Trigger
<b>Syntax</b>	TRIGger:CLKRec:CRC:LOCKState?
<b>Returns</b>	May return UNLOCKed, ACQuiringlock, LOCKed, or HIGHJitter.
<b>Examples</b>	TRIGGER:CLKREC:CRC:LOCKSTATE? might return :TRIGGER:CLKREC:CRC:LOCKSTATE LOCK indicating that the clock recovery instrument is locked to the signal.

---

**NOTE.** UNLOCKED is returned if a clock recovery instrument is not present.

---

## TRIGger:CLKRec:CRC:LOOPBandwidth

This command sets or queries the loop bandwidth of the clock recovery instrument.

<b>Group</b>	Trigger
<b>Syntax</b>	TRIGger:CLKRec:CRC:LOOPBandwidth <NR3> TRIGger:CLKRec:CRC:LOOPBandwidth?
<b>Arguments</b>	NR3 sets the loop bandwidth. The range is 100E3 to 12E6.
<b>Examples</b>	TRIGGER:CLKREC:CRC:LOOPBANDWIDTH 100E4 sets the loop bandwidth to 1 MHz.  TRIGGER:CLKREC:CRC:LOOPBANDWIDTH? might return :TRIGGER:CLKREC:CRC:LOOPBANDWIDTH 7.1980000000E6, indicating that the loop bandwidth is set to 7.198 MHz.

---

**NOTE.** NAN is returned if a clock recovery instrument is not present.

---

## TRIGger:CLKRec:CRC:MODEInum? (Query Only)

Returns a string with the clock recovery instrument information.

<b>Group</b>	Trigger
--------------	---------

**Syntax** TRIGger:CLKRec:CRC:MODELnum?

**Examples** TRIGGER:CLKREC:CRC:MODELNUM? might return BERTScope DCRJ, indicating that the current clock recovery instrument is a BERTScope that supports jitter analysis.

## TRIGger:CLKRec:CRC:NOMEDgedensity

This command sets or queries the nominal edge density (in percent) of the clock recovery instrument. The edge density refers to the average percentage of unit intervals which contain a transition.

**Group** Trigger

**Syntax** TRIGger:CLKRec:CRC:NOMEDgedensity <NR3>  
TRIGger:CLKRec:CRC:NOMEDgedensity?

**Related Commands** [TRIGger:CLKRec:CRC:EDGE DENSITYMode](#)

**Arguments** NR3 sets the edge density value. The range is from 10% to 100%

**Examples** TRIGGER:CLKREC:CRC:NOMEDGE DENSITY 50 sets the edge density value to 50%.

TRIGGER:CLKREC:CRC:NOMEDGE DENSITY? might return  
:TRIGGER:CLKREC:CRC:NOMEDGE DENSITY 50.0000000000,  
indicating that the current edge density is set to 50%.

---

**NOTE.** NAN is returned if a clock recovery instrument is not present.

---

## TRIGger:CLKRec:CRC:NOMFrequency

This command sets or queries the nominal frequency setting (in Hz) of the clock recovery instrument.

**Group** Trigger

**Syntax** TRIGger:CLKRec:CRC:NOMFrequency <NR3>  
TRIGger:CLKRec:CRC:NOMFrequency?

- Arguments** NR3 is the nominal frequency value. The frequency range is 150 MHz to 12.5 GHz.
- Examples** TRIGGER:CLKREC:CRC:NOMFREQUENCY 12.5E9 sets the nominal frequency of the clock recovery instrument to 12.5 GHz.
- TRIGGER:CLKREC:CRC:NOMFREQUENCY? might return  
:TRIGGER:CLKREC:CRC:NOMFREQUENCY 12.0000000000E9,  
indicating that the nominal frequency of the clock recovery instrument is set to 12 GHz.

---

**NOTE.** *NAN is returned if a clock recovery instrument is not present.*

---

## TRIGger:CLKRec:CRC:PEAKing

This command sets or queries the peak range (in dB) of the clock recovery instrument.

- Group** Trigger
- Syntax** TRIGger:CLKRec:CRC:PEAKing <NR3>  
TRIGger:CLKRec:CRC:PEAKing?
- Arguments** NR3 sets the peak range from 0 dB to 6 dB.
- Examples** TRIGGER:CLKREC:CRC:PEAKING 3 dB sets the peak range to 3 dB.
- TRIGGER:CLKREC:CRC:PEAKING? might return  
:TRIGGER:CLKREC:CRC:PEAKING 2.1000000000, indicating that peaking range is set to 2.1 dB.

---

**NOTE.** *NAN is returned if a clock recovery instrument is not present.*

---

## TRIGger:CLKRec:CRC:PHASEERRLimit

This command sets or queries the phase error limit (in percent unit intervals) of the clock recovery instrument. This is the peak-to-peak phase error between the incoming data and the recovered clock.

- Group** Trigger

**Syntax** TRIGger:CLKRec:CRC:PHASEERRLimit <NR3>  
TRIGger:CLKRec:CRC:PHASEERRLimit?

**Arguments** NR3 selects a range from 10% UI to 90% UI.

**Examples** TRIGGER:CLKREC:CRC:PHASEERRLIMIT 50 sets the phase error limit to 50%.  
TRIGGER:CLKREC:CRC:PHASEERRLIMIT? might return  
:TRIGGER:CLKREC:CRC:PHASEERRLIMIT 50.0000000000,  
indicating that phase error limit is set to 50% UI.

---

**NOTE.** *NAN is returned if a clock recovery instrument is not present.*

---

## TRIGger:CLKRec:CRC:PHASEERRP2P? (Query Only)

This query only command returns the phase error peak-to-peak measurement (in percent unit intervals) from the clock recovery instrument.

**Group** Trigger

**Syntax** TRIGger:CLKRec:CRC:PHASEERRP2P?

**Examples** TRIGGER:CLKREC:CRC:PHASEERRP2P? might return  
:TRIGGER:CLKREC:CRC:PHASEERRP2P 100.0000000000, indicating that  
the peak-to-peak phase error is 100%.

---

**NOTE.** *NAN is returned if a clock recovery instrument is not present.*

---

## TRIGger:CLKRec:CRC:PHASEERRRMS? (Query Only)

This query only command returns the phase error RMS measurement (in percent unit intervals) from the clock recovery instrument.

**Group** Trigger

**Syntax** TRIGger:CLKRec:CRC:PHASEERRRMS?

**Examples** TRIGGER:CLKREC:CRC:PHASEERRRMS? might return  
:TRIGGER:CLKREC:CRC:PHASEERRRMS 22.55348205566, indicating that the  
RMS phase error is 22.55%.

---

**NOTE.** *NAN is returned if a clock recovery instrument is not present.*

---

## TRIGger:CLKRec:CRC:RCONfigdevice (No Query Form)

This command restores the clock recovery instrument to the settings from one of the non-volatile setup locations in the clock recovery instrument.

**Group** Trigger

**Syntax** TRIGger:CLKRec:CRC:RCONfigdevice {POWER\_ON | SETUP\_1 |  
SETUP\_2 | SETUP\_3 | SETUP\_4 | FACTORY}

**Related Commands** [TRIGger:CLKRec:CRC:SCONfigdevice](#)

- Arguments**
- POWER\_ON restores the instrument to the configuration at power on.
  - SETUP\_1 restores the instrument to the settings saved in memory location 1 of the clock recovery instrument.
  - SETUP\_2 restores the instrument to the settings saved in memory location 2 of the clock recovery instrument.
  - SETUP\_3 restores the instrument to the settings saved in memory location 3 of the clock recovery instrument.
  - SETUP\_4 restores the instrument to the settings saved in memory location 4 of the clock recovery instrument.
  - FACTORY restores the instrument to the factory default configuration.

**Examples** TRIGGER:CLKREC:CRC:RCONFIGDEVICE FACTORY returns the clock recovery instrument to the factory default configuration.

## TRIGger:CLKRec:CRC:RELock (No Query Form)

This command causes the clock recovery instrument to search for the data rate.

**Group** Trigger

**Syntax** TRIGger:CLKRec:CRC:RELock

**Related Commands** [TRIGger:CLKRec:CRC:LOCKState?](#)

**Examples** TRIGGER:CLKREC:CRC:RELOCK causes the connected clock recovery instrument to search and lock to the data rate.

## TRIGger:CLKRec:CRC:RESETLockcount (No Query Form)

This command resets the lock count to zero on the clock recovery instrument.

**Group** Trigger

**Syntax** TRIGger:CLKRec:CRC:RESETLockcount

**Related Commands** [TRIGger:CLKRec:CRC:RELock](#)

**Examples** TRIGGER:CLKREC:CRC:RESETLOCKCOUNT resets the lock count back to zero on the connected clock recovery instrument.

## TRIGger:CLKRec:CRC:SCONfigdevice (No Query Form)

This command saves the current settings of the clock recovery instrument to one of the non-volatile setup locations in the clock recovery instrument.

**Group** Trigger

**Syntax** TRIGger:CLKRec:CRC:SCONfigdevice {POWER\_ON | SETUP\_1 | SETUP\_2 | SETUP\_3 | SETUP\_4}

**Related Commands** [TRIGger:CLKRec:CRC:RCONfigdevice](#)



- Arguments**
- POWER\_ON is the configuration used at power on.
  - SETUP\_1 is memory location 1.
  - SETUP\_2 is memory location 2.
  - SETUP\_3 is memory location 3.
  - SETUP\_4 is memory location 4.

**Examples** TRIGGER:CLKREC:CRC:SCONFIGDEVICE SETUP\_1 stores the current clock recovery instrument instrument settings to memory location 1.

## TRIGger:CLKRec:CRC:SERIalnum? (Query Only)

This query only command returns the serial number of the clock recovery instrument.

**Group** Trigger

**Syntax** TRIGger:CLKRec:CRC:SERIalnum?

**Examples** TRIGGER:CLKREC:CRC:SERIALNUM? might return  
:TRIGGER:CLKREC:CRC:SERIALNUM B010112.

---

**NOTE.** *undefined is returned if a clock recovery instrument is not present.*

---

## TRIGger:CLKRec:CRC:STANdard

This command sets or queries the communications standard of the clock recovery instrument. Selecting a new standard changes the clock recovery instrument settings for NOMFREQUENCY, LOCKRANGE, LOOPBANDWIDTH, PEAKING, and NOMEDGEDENSITY.

**Group** Trigger

**Syntax** TRIGger:CLKRec:CRC:STANdard <QString>  
TRIGger:CLKRec:CRC:STANdard?

**Arguments** QString is the names of the clock recovery standard.

- Examples** TRIGGER:CLKREC:CRC:STANDARD "GBE10" sets the clock recovery standard of the clock recovery instrument to GBE10.
- TRIGGER:CLKREC:CRC:STANDARD? might return  
:TRIGGER:CLKREC:CRC:STANDARD OC12 indicating that the currently selected standard of the clock recovery instrument is OC12.

## TRIGger:CLKRec:CRC:STANDARDList? (Query Only)

This query only command returns a comma-separated list of the standards available on the clock recovery instrument.

- Group** Trigger
- Syntax** TRIGger:CLKRec:CRC:STANDARDList?
- Examples** TRIGGER:CLKREC:CRC:STANDARDLIST? might return  
:TRIGGER:CLKREC:CRC:STANDARDLIST SATA1(0.9), SATA1(1.98), ...  
indicating these are the standards supported by the connected clock recovery instrument.

---

**NOTE.** *undefined is returned if a clock recovery instrument is not present.*

---

## TRIGger:CLKRec:CRC:SUBCLOCKAmpl

This command sets or queries the sub-rate clock output amplitude (in mV) of the clock recovery instrument.

- Group** Trigger
- Syntax** TRIGger:CLKRec:CRC:SUBCLOCKAmpl <NR3>  
TRIGger:CLKRec:CRC:SUBCLOCKAmpl?
- Arguments** NR3 sets the sub-rate clock output amplitude. The range is 250 mV to 900 mV.
- Examples** TRIGGER:CLKREC:CRC:SUBCLOCKAMPL 300E-3 sets the sub-rate clock output amplitude to 300 mV.

TRIGGER:CLKREC:CRC:SUBCLOCKAMPL? might return  
:TRIGGER:CLKREC:CRC:SUBCLOCKAMPL 0.5000000000,  
indicating that the sub-rate clock output amplitude is 500 mV.

---

**NOTE.** *NAN is returned if a clock recovery instrument is not present.*

---

## TRIGger:CLKRec:CRC:SUBCLOCKDiv

This command sets or queries the division ratio of the sub-rate clock of the clock recovery instrument.

**Group** Trigger

**Syntax** TRIGger:CLKRec:CRC:SUBCLOCKDiv <NR1>  
TRIGger:CLKRec:CRC:SUBCLOCKDiv?

**Arguments** ■ <NR1>.

The available sub-rate clock divisors include:

1, 2, 4, 5, 6, 7, 8, 9, 10, 12, 14, 16, 18, 20, 24, 25, 28, 30, 32, 35, 36, 40, 42, 45,  
48, 49, 50, 54, 56, 60, 63, 64, 70, 72, 80, 81, 90, 100, 108, 112, 120, 126, 128,  
140, 144, 160, 162, 168, 180, 192, 196, 200, 216, 224, 240, 252, 256, 280,  
288, 320, 324, 336, 360, 384, 392, 432, 448, 504, 512, 576, 648

Setting the divisor to 1 equals full rate.

**Examples** TRIGGER:CLKREC:CRC:SUBCLOCKDIV 2 sets the sub-rate clock output to one  
half the clock rate.

TRIGGER:CLKREC:CRC:SUBCLOCKDIV? might return  
:TRIGGER:CLKREC:CRC:SUBCLOCKDIV 4 , indicating that the sub-rate clock  
output is one fourth the rate of the clock.

---

**NOTE.** *1 is returned if a clock recovery instrument is not present.*

---

## TRIGger:CLKRec:CRC:SUBCLOCKDIVList? (Query Only)

This query only command returns a comma-separated list of available sub-rate  
clock divisors of the clock recovery instrument.

**Group** Trigger

**Syntax** TRIGger:CLKRec:CRC:SUBCLOCKDIVList?

**Examples** TRIGGER:CLKREC:CRC:SUBCLOCKDIVLIST? might  
return :TRIGGER:CLKREC:CRC:SUBCLOCKDIVLIST  
1,2,4,5,6,7,8,9,10,12,14,16,18,20,24,25,28,30,32,35,36,40,42,45.

---

**NOTE.** *undefined is returned if a clock recovery instrument is not present.*

---

## TRIGger:CLKRec:CRC:SUBCLOCKOutput

This command sets or queries the state of the sub-rate clock output of the clock recovery instrument.

**Group** Trigger

**Syntax** TRIGger:CLKRec:CRC:SUBCLOCKOutput {ON | OFF | <NR1>}  
TRIGger:CLKRec:CRC:SUBCLOCKOutput?

- Arguments**
- ON specifies that the clock recovery instrument sub-rate clock output is enabled.
  - OFF specifies that the clock recovery instrument sub-rate clock output is disabled.
  - <NR1> 1 enables the sub-rate clock output. 0 disables the sub-rate clock output.

**Examples** TRIGGER:CLKREC:CRC:SUBCLOCKOUTPUT ON enables the sub-rate clock output on the clock recovery instrument.

TRIGGER:CLKREC:CRC:SUBCLOCKOUTPUT? might return  
:TRIGGER:CLKREC:CRC:SUBCLOCKOUTPUT ON, indicating that the  
sub-rate clock output is enabled on the clock recovery instrument.

---

**NOTE.** *0 is returned if a clock recovery instrument is not present.*

---

## TRIGger:CLKRec:CRC:SWRev? (Query Only)

This query only command returns the firmware revision string of the clock recovery instrument.

<b>Group</b>	Trigger
<b>Syntax</b>	TRIGger:CLKRec:CRC:SWRev?
<b>Examples</b>	TRIGGER:CLKREC:CRC:SWREV? might return :TRIGGER:CLKREC:CRC:SWREV XXXX (XXXX represents the firmware version).

---

**NOTE.** *UNDEFINED* is returned if a clock recovery instrument is not present.

---

## TRIGger:CLKRec:SOUrce

This command sets or queries the clock recovery trigger source. The clock recovery trigger source is derived from either the optical input of an 80C0x optical module with an installed clock recovery option, or a clock recovery accessory module (such as the 80A05). This clock is synchronous with the incoming data signal. Use of this trigger source requires a module that supports clock recovery. The module picks the clock off internally and outputs it to the instrument for triggering.

<b>Group</b>	Trigger
<b>Syntax</b>	TRIGger:CLKRec:SOUrce {CH<x>   NONE} TRIGger:CLKRec:SOUrce?
<b>Arguments</b>	<ul style="list-style-type: none"> <li>■ CH&lt;x&gt; is the channel for which you want to set the clock recovery source. The value for x can be 1, 2, 3, or 4.</li> <li>■ NONE. Setting or querying this argument indicates that the specified channel has no available clock recovery options. You cannot set a specified channel to NONE if there are available clock recovery options.</li> </ul>
<b>Examples</b>	TRIGGER:CLKREC:SOURCE CH2 sets the source for clock recovery to Channel 2.  TRIGGER:CLKREC:SOURCE? might return TRIGGER:CLKREC:SOURCE CH1, indicating that Channel 1 is the source for clock recovery.

## TRIGger:HIFreq

This command sets or queries whether the high-frequency triggering function is enabled. The enhanced triggering control increases the trigger sensitivity of the trigger circuit by decreasing hysteresis. This allows triggering on higher frequency

signals. This command is equivalent to selecting High Frequency Triggering in the Enhanced Triggering section of the Trigger Setup dialog box. High Frequency Triggering is enabled only when the trigger source is External Direct.

**Group** Trigger

**Syntax** TRIGGER:HIFreq { ON | OFF | NR1 }  
TRIGGER:HIFreq?

**Related Commands** [TRIGGER:SOURce](#)

- Arguments**
- ON enables high-frequency triggering.
  - OFF disables high-frequency triggering.
  - NR1 set to 0 disables the high-frequency triggering; any other value enables the high-frequency triggering.

**Examples** TRIGGER:HIFREQ OFF disables the high-frequency triggering.  
TRIGGER:HIFREQ? might return TRIGGER:HIFREQ 1, indicating that the high-frequency triggering is enabled.

---

**NOTE.** *A query always returns a 0 or 1.*

---

## TRIGGER:HOLDoff

This command sets or queries the trigger holdoff, which is the time the instrument waits before arming the trigger system to accept triggers. Holdoff can help achieve stable triggering. Sending this command is equivalent to entering a value in the Holdoff control in the Trigger Setup dialog box.

Two primary factors determine what the minimum acceptable holdoff value may be at any given time.

- **Hardware restrictions.** There is a minimum time required from the acceptance of one trigger event until the next due to both trigger re-arm time and maximum sample rate requirements. These requirements result in an absolute minimum holdoff setting of 5  $\mu$ s (given appropriate instrument settings).
- **Time of last sample.** The instrument must use a holdoff setting that is at least greater than the time from the trigger event to the farthest sample in any waveform acquisition. For example, if you have the time/div set to 1  $\mu$ s/div with the horizontal timebase position set to minimum, for instance about 18 ns, the operational (that is, actual) holdoff must be greater than: 19 ns + 1  $\mu$ s/div\*10 div = 10.018  $\mu$ s (the hardware restrictions cause the instrument to increase this value).

The following are many of the parameters, which affect the determination by the instrument of the minimum actual holdoff:

- Horizontal scale (time/div)
- Horizontal position
- Horizontal FrameScan settings (when FrameScan is enabled)
- User (Channel) deskew

The query form of this command returns the actual holdoff value in use by the instrument.

<b>Group</b>	Trigger
<b>Syntax</b>	TRIGger:HOLDoff <NR3> TRIGger:HOLDoff?
<b>Related Commands</b>	<a href="#">HORizontal:MAIn:SCAle</a> , <a href="#">HORizontal:MAIn:POSition</a> , <a href="#">CH&lt;x&gt;:DESkew</a>
<b>Arguments</b>	NR3 is the value of the holdoff time.
<b>Examples</b>	TRIGGER:HOLDOFF 5.0E-6 sets the requested trigger holdoff to 5 $\mu$ s. TRIGGER:HOLDOFF? might return TRIGGER:HOLDOFF 6.0000000000E-006, indicating that the actual holdoff is set to 6 $\mu$ s.

## TRIGger:INTRate

This command sets or queries the TDR (internal clock) rate. The command sets the same parameter as the TDR:INTRate command. The instrument uses the internal

clock rate to synchronize the generation of TDR step pulses with the subsequent acquisition. The internal clock in the instrument generates a continuous signal with selectable repetition rates of from 25 to 300 kHz in 1 kHz increments. The actual repetition rate may vary by as much as 4% from the specified value.

Sending this command is equivalent to selecting a value for Internal Clock in the TDR (or Trigger) Setup dialog box.

<b>Group</b>	TDR
<b>Syntax</b>	TRIGger:INTRate <NR3> TRIGger:INTRate?
<b>Arguments</b>	NR3 sets the internal clock rate to any value in the range of 25 kHz to 300 kHz. The actual TDR repetition rate may vary by up to 4% from the specified value. You can input any NR3 value; the instrument will set the internal clock rate to the closest available setting.
<b>Examples</b>	TRIGGER:INTRATE 100E3 sets the internal clock rate to 100 kHz.  TRIGGER:INTRATE? might return TRIGGER:INTRATE 0.200000000000E+006, indicating that the internal clock rate is set to 200 kHz (the default value).

## TRIGger:LEVel

This command queries or sets the trigger level, which is applied only when the trigger source is set to External Direct. The trigger level is the voltage threshold through which the trigger source signal must pass to generate a trigger event. This command is equivalent to setting a value for the level in the Level section of the Trigger Setup dialog box or adjusting the Trigger LEVEL control on the front panel.

<b>Group</b>	Trigger
<b>Syntax</b>	TRIGger:LEVel <NR3> TRIGger:LEVel?
<b>Related Commands</b>	<a href="#">TRIGger:SOUrce</a> , <a href="#">TRIGger:SETLevel</a>
<b>Arguments</b>	NR3 is the value of the trigger level in volts.



**Examples** TRIGGER:LEVEL 1.0E-1 sets the trigger level to 100 mV.  
 TRIGGER:LEVEL? might return TRIGGER:LEVEL 0.24100000000, indicating that the trigger level is set to 241 mV.

## TRIGger:MODE

This command sets or queries the trigger mode. This is equivalent to selecting a mode in the Mode section of the Trigger Setup dialog box. These modes apply only when the Trigger Source is set to Clock/Prescale, C1 clock recovery or C3 clock recovery.

Select Eye Mode if the signal you are acquiring is synchronous with the selected clock source and you wish to display the acquired data as an "eye diagram" overlaying all of the acquired data bits with minimal clock to data jitter.

Select Pattern mode if the signal you are acquiring is synchronous with the selected clock source and you wish to display the individual bits as a "bit stream", with minimal clock to data jitter.

---

**NOTE.** *Pattern mode is only available if the DSA8300 has the ADVTRIG option present.*

---

Select Other mode if the signal you are acquiring is synchronous with the selected clock source and you wish to display the acquired data showing the effects of any clock modulation (such as Spread Spectrum Clocking).

**Group** Trigger

**Syntax** TRIGger:MODE {EYE | OTHER| PATtern}  
 TRIGger:MODE?

**Related Commands** [TRIGger:SOURce](#)

- Arguments**
- EYE sets the trigger mode to Eye (see above description).
  - OTHER sets the trigger mode to Other (see above description).
  - PATtern sets the trigger mode to Pattern (see above description).

**Examples** TRIGGER:MODE EYE sets the trigger mode to Eye.  
 TRIGGER:MODE? might return TRIGGER:MODE PATTERN, indicating that the instrument is set to acquire pattern synchronous data if a clock source is selected for the Trigger Source,

## TRIGger:PSYNc:DATARate

This command sets or queries the data rate (<nr3>) of the PatternSync portion of the trigger system. The data rate must be in the range valid for the clock trigger source selected.

The query form returns the current data rate for the PatternSync Trigger portion of the trigger system.

If you attempt to modify this parameter while the AUTOSync:DATARate is ON, you will receive Execution Error code 2539 with an error message "Pattern Sync Parameter is not settable". The *Messages* section contains general information on programming interface messages.(See page 3-11.)

**Group** Trigger

**Syntax** TRIGger:PSYNc:DATARate <nr3>  
 TRIGger:PSYNc:DATARate?

**Arguments** <nr3> sets the data rate in bits per second.

**Examples** TRIGGER:PSYNc:DATARATE10.3125E9 sets the data rate to 10.3125 Gb/s.  
 TRIGGER:PSYNc:DATARATE? might return TRIGGER:PSYNc:DATARATE 9.95328000E9, indicating that the data rate is 9.95328 Gb/s.

## TRIGger:PSYNc:DCRAtio

This command sets or queries the data-to-clock ratio (<nr1>,<nr1>) of the PatternSync Trigger portion of the trigger system. The first <nr1> value represents the data rate and the second <nr1> value represents the clock rate.

One of the data values in the Data/Clock ratio pair must be a 1. Valid data entries range from 1 to 128. A constraint on the valid range is set by the Pattern Length and Data Rate settings. Invalid entries are ignored, leaving the ratio unchanged.

The query form returns the current data-to-clock ratio.

If you attempt to modify this parameter while the AUTOSync:DCRAtio is ON, you will receive Execution Error code 2539 with an error message "Pattern Sync Parameter is not settable". The *Messages* section contains general information on programming interface messages.(See page 3-11.)

<b>Group</b>	Trigger
<b>Syntax</b>	TRIGger:PSync:DCRAtio <nr1>,<nr1> TRIGger:PSync:DCRAtio?
<b>Arguments</b>	<nr1> (first argument) sets the data rate. <nr1> (second argument) sets the clock rate.
<b>Examples</b>	TRIGGER:PSync:DCRATIO2,1 sets the data-to-clock ratio as 2:1. TRIGGER:PSync:DCRATIO? might return TRIGGER:PSync:DCRATIO 1,4, indicating that the current data-to-clock ratio is 1:4.

## TRIGger:PSync:PLENgtH

This command sets or queries the pattern length (<nr1>) of the PatternSync Trigger portion of the trigger system. The pattern length must be in the range of 2 to 2<sup>23</sup> (8,388,608)

The query form returns the current pattern length.

If you attempt to modify this parameter while the AUTOSync:PLENgtH is ON, you will receive Execution Error code 2539 with an error message "Pattern Sync Parameter is not settable". The *Messages* section contains general information on programming interface messages.(See page 3-11.)

<b>Group</b>	Trigger
<b>Syntax</b>	TRIGger:PSync:PLENgtH <nr1> TRIGger:PSync:PLENgtH?
<b>Arguments</b>	<nr1> sets the pattern length in bits.
<b>Examples</b>	TRIGGER:PSync:PLENGTH 127 sets the pattern sync pattern length to 127 bits. TRIGGER:PSync:PLENGTH? might return TRIGGER:PSync:PLENGTH 32767, indicating that the pattern length is 32,767 bits.

## TRIGger:SETLevel (No Query Form)

This command (no query form) sets the trigger level automatically to 50% of the minimum/maximum value of the trigger input signal. This trigger level is effective only when the trigger source is set to External Direct (see the [TRIGger:SOUrce](#) command). The trigger level is the voltage threshold through which the trigger source signal must pass to generate a trigger event. This command is equivalent to clicking Set to 50% in the Level section of the Trigger Setup dialog box or pressing the Trigger Set to 50% button on the front panel.

**Group** Trigger

**Syntax** TRIGger:SETLevel

**Related Commands** [TRIGger:SOUrce](#), [TRIGger:LEVel](#)

**Examples** TRIGGER:SETLEVEL sets the trigger level to 50% of the applied signal.

## TRIGger:SLOpe

This command sets or queries the trigger slope. The slope is applied only when trigger source is set to External Direct (see the [TRIGger:SOUrce](#) command). For Clock, TDR, and Free Run trigger sources, the slope is set internally to rising. Sending this command is equivalent to selecting the rising or falling icon in the Slope section of the Trigger Setup dialog box.

**Group** Trigger

**Syntax** TRIGger:SLOpe {RISe | FALL}  
TRIGger:SLOpe?

**Related Commands** [TRIGger:SOUrce](#)

- Arguments**
- RISe causes the instrument to trigger on the rising slope of the trigger source signal.
  - FALL causes the instrument to trigger on the falling slope of the trigger source signal.

- Examples** TRIGGER:SLOPE FALL causes the instrument to trigger on the falling slope of the trigger source signal when the trigger source is set to External Direct.
- TRIGGER:SLOPE? might return TRIGGER:SLOPE RISE, indicating that the instrument is set to trigger on the rising slope of the trigger source signal.

## TRIGger:SOURce

This command sets or queries the trigger source, which provides the signal that triggers acquisition. This is equivalent to selecting one of the available trigger sources in the Trigger Source section of the Trigger Setup dialog box.

The available trigger sources are:

- C1CLKRec. xxx
- C3CLKRec. xxx
- External Direct. Select this trigger source xxxxx
- External Prescaler. Select this trigger source xxx
- Freerun. xx
- Internal Clock. Select this trigger source xxx
- TDR. Select this trigger source xxx

**Group** Trigger

**Syntax** TRIGger:SOURce { C1CLKRec | C3CLKRec | EXTDirect |  
EXTPrescaler | FREerun | INTClk | TDR }  
TRIGger:SOURce?

**Related Commands** [TRIGger:CLKRec:SOURce](#)

- Arguments**
- **C1CLKRec** sets the trigger source to the clock recovery output of the module in the upper left slot. This trigger source requires either an optical sampling head that supports clock recovery or the 80A05 clock recovery module.
  - **C3CLKRec** sets the trigger source to the clock recovery output of the module in the upper right slot. This trigger source requires either an optical sampling head that supports clock recovery or the 80A05 clock recovery module.
  - **EXTDirect** sets the instrument to trigger on an external signal connected to the Trigger Direct input connector on the instrument front panel. The input coupling mode is DC and triggering is usable with signals up to at least 3.0 GHz. Use this trigger source for low repetition rate signals that cannot be triggered using the Clock Input/Prescale Trigger front panel connector.
  - **EXTPrescaler** sets the instrument to trigger on an external signal connected to the Clock Input/Prescale Trigger input connector on the instrument front panel. The input coupling is AC. Triggering is guaranteed for clocks up to 15 GHz (and will typically work up to 20 GHz).
  - **FREerun** sets the instrument to disable synchronous triggered acquisition; the instrument randomly triggers to acquire data.
  - **INTClk** sets the instrument to trigger on an internal clock signal. The INTClk trigger source is a synonym for TDR and is provided for backwards compatibility with legacy 8000 series sampling oscilloscopes. Setting the Trigger Source to INTClk and then sending the TRIGGER:SOURCE? returns the "TDR" synonym.
  - **TDR** sets the instrument to trigger on an internal clock signal. Use the TRIGGER:INTRate command to select to select any rate between 25 kHz and 300 kHz. Use this type of trigger source with TDR to synchronize the generation of TDR step pulses with the subsequent acquisition. The Internal Clock Out connector on the instrument front panel supplies a sample of the internal clock.

**Examples** TRIGGER:SOURCE EXTDIRECT sets the trigger source to use the Trigger Direct Input front panel connector.

TRIGGER:SOURCE? might return EXTPrescaler, indicating that the trigger source is currently set to the Clock Input / Prescale Trigger front panel connector.

## TRIGger:STATE? (Query Only)

This query returns the status of the trigger system.

**Group** Trigger

**Syntax** TRIGger:STATE?

**Related Commands** [TRIGger:MODE](#)

**Returns** TRIGGERED indicates that the trigger system is active, and the instrument is triggered.

NOTTRIGGERED indicates that the trigger system is active, and the instrument is not triggered.

STOPPED indicates that the acquisition is disabled so the state of trigger system is ignored.

**Examples** TRIGGER:STATE? might return TRIGGER:STATE TRIGGERED, indicating that the instrument is triggered.

## UNLock (No Query Form)

This command unlocks the front panel and touchscreen. This command is equivalent to LOCK NONE. There is no query form for this command.

---

**NOTE.** *If the instrument is in the Remote With Lockout State (RWLS), the UNLOCK command has no effect. For more information see the ANSI-IEEE Std. 488.1-1987 Standard Digital Interface for Programmable Instrumentation, section 2.8.3 on RL State Descriptions.*

---

**Group** Miscellaneous

**Syntax** UNLOCK ALL

**Related Commands** [LOCK](#)

**Arguments** ALL unlocks all front-panel buttons, knobs and the touchscreen.

**Examples** UNLOCK ALL unlocks the front panel and touchscreen

## VERBoSe

This command sets or queries the verbose state that controls the length of keywords on query responses. Keywords can be both headers and arguments. This command does not affect IEEE Std 488.2-1987 Common Commands (those starting with an asterisk).

**Group** Miscellaneous

**Syntax** VERBoSe {OFF | ON | <NR1>}  
VERBoSe?

**Related Commands** [HEADer](#), [\\*LRN?](#), [SET?](#)

- Arguments**
- ON sets the verbose state true, which returns full-length keywords for applicable setting queries.
  - OFF sets the verbose state false, which returns minimum-length keywords for applicable setting queries.
  - NR1 set to 0 sets the verbose state false, which returns minimum-length keywords for applicable setting queries; any other integer sets the verbose state true, which returns full-length keywords for applicable setting queries. A query always returns a 0 or 1.

**Examples** VERBOSE ON sets the verbose state true, which returns full-length keywords for queries.

VERBOSE? might return VERBOSE 1, indicating that the verbose state is true.

---

**NOTE.** *A query always returns a 0 or 1.*

---

## \*WAI (No Query Form)

The \*WAI (Wait) command (no query form) prevents the instrument from executing further commands or queries until all pending operations finish. This command allows you to synchronize the operation of the instrument with your application program. For more information, refer to Synchronization Methods.

**Group** Status and Error

**Syntax** \*WAI



**Related Commands** [BUSY?](#), [\\*OPC](#)

**Examples** \*WAI prevents the instrument from executing any further commands or queries until all pending commands that generate an OPC message are complete.

## WAVFrm? (Query Only)

This is a query only command that returns CURVe? and WFMOUpre? data (data and outgoing waveform preamble) for the waveform as specified by the DATA:SOURce command. This command is equivalent to sending the CURVe? and WFMOUpre? commands.

**Group** Waveform Transfer

**Syntax** WAVFrm?

**Related Commands** [CURVe](#), [DATA:SOURce](#), [DATA:TIMEbase](#), [WFMOUpre?](#)

**Examples** WAVFRM? might return (partial) WAVFRM  
 -447610880,-467206144,-501547008,  
 -499187712,-513605632,-510001152,-511508480,  
 -519241728,-504365056,-504365056,-507576320,  
 -509214720,-509476864,-509476864,-509476864,  
 -505348096,-508559360,-511639552,-509739008,  
 -510984192,-513015808,-509214720,-510001152,  
 -508362752,-512688128,-512950272,-509083648,  
 -510328832,-511705088,-510197760,-508166144,  
 -506265600,-514588672,...;,:WFMOUPRE:ENCDG ASCII; BN\_FMT  
 RI;BYT\_OR LSB;BYT\_NR 4;BIT\_NR 32; NR\_PT 500;PT\_FMT Y;XINCR  
 500.000000000000E-012; XZERO 18.000000000008E-009;XUNIT  
 ""; YMULT 0.100000000000;XMULT 500.00000000000E-012;  
 XOFF 20.500000000008E-009;YZERO 0.000000000000; YSCALE  
 232.83064365443E-012;YOFF 0;YUNIT ""; WFID "PULSE TEST,  
 100.0MV/DIV, 500.0PS/DIV, 500 POINTS, SAMPLE MODE";WFMLABEL  
 "PULSE TEST" ...

## WFMDB? (Query Only)

This query only command returns all the waveform database parameters.

**Group** Waveform Database

**Syntax**      WFMDB?

**Related Commands**      [WFMDB:DISPlay?](#)

**Examples**      WFMDB? might return WFMDB:DISPLAY:EMPHASIS 50;GRADING COLOR;INVERT 0;GRADMETHOD EMPH8;:WFMDB:WFMDB1 SOURCE CH2,MAIN;DISPLAY 1;ENABLE 1;PERSISTENCE:MODE VARPERSIST;COUNT 2000;SAMPLES 500000;:WFMDB:WFMDB2 SOURCE CH4,MAIN;DISPLAY 1;ENABLE 1;PERSISTENCE MODE VARPERSIST;COUNT 2000;SAMPLES 500000;:WFMDB WFMDB3:SOURCE CH3,MAIN;DISPLAY 1;ENABLE 1; PERSISTENCE:MODE INFERSIST;COUNT 500;SAMPLES 125000; WFMDB:WFMDB4:SOURCE MATH3,MAG1;DISPLAY 0;ENABLE 0; PERSISTENCE:MODE INFERSIST;COUNT 500;SAMPLES 125000.

## WFMDB:DISPlay?P (Query Only)

This query only command returns all the WfmDB display parameters.

**Group**      Waveform Database

**Syntax**      WFMDB:DISPlay?

**Related Commands**      [WFMDB?](#)

**Examples**      WFMDB:DISPLAY? might return WFMDB:DISPLAY:EMPHASIS 50;GRADING COLOR;INVERT 0.

## WFMDB:DISPlay:EMPHAsis

This command sets or queries the display emphasis for the WfmDB (waveform database). The value you specify sets the range of counts that you want to emphasize. The lowest value, 0, emphasizes bins with low counts; the highest value, 100, emphasizes bins with high counts. You can set the emphasis from 0% to 100%. This command is the equivalent to using the slide bar to set the Emphasize Counts percentage in the Display Options section of the WfmDB Setup dialog box.

**Group**      Waveform Database

**Syntax** `WFMDB:DISPlay:EMPHasis <NR3>`  
`WFMDB:DISPlay:EMPHasis?`

**Related Commands** [WFMDB:DISPlay:GRADing](#), [WFMDB:DISPlay:INVert](#)

**Arguments** NR3 is the percentage to which Emphasize Counts is set. The range is from 0 through 100.

**Examples** `WFMDB:DISPLAY:EMPHASIS 0` sets the instrument to emphasize in the display bins with low counts.

`WFMDB:DISPLAY:EMPHASIS?` might return `WFMDB:DISPLAY:EMPHASIS 50`, indicating that the Emphasize Counts is set to 50%.

## WFMDB:DISPlay:GRADing

This command sets or queries which grading type (color or intensity) the instrument uses to draw the WfmDB. This is equivalent to selecting Color or Intensity for Grading in the Display Options section of the WfmDB Setup dialog box.

**Group** Waveform Database

**Syntax** `WFMDB:DISPlay:GRADing {COLor | INTensity}`  
`WFMDB:DISPlay:GRADing?`

**Related Commands** [WFMDB:DISPlay:EMPHasis](#), [WFMDB:DISPlay:INVert](#)

**Arguments**

- `COLor` maps the WfmDB display to colors. This is the default setting.
- `INTensity` maps the WfmDB display to intensity.

**Examples** `WFMDB:DISPLAY:GRADING COLOR` sets the grading type for the WfmDB to color.

`WFMDB:DISPLAY:GRADING?` might return `WFMDB:DISPLAY:GRADING INTENSITY`, indicating that the grading type for the WfmDB display is set to intensity.

## WFMDB:DISPlay:GRADMethod

This command sets or queries the grading method that waveform databases use to display the data they contain. For example, setting an emphasis mode displays data so it is graded by count frequencies in respect to the bin levels in which the data accumulates.

Use of this command is equivalent to selecting Grading Method in the Display Options section of the Wfm Database Setup dialog box.

**Group** Waveform Database

**Syntax** WFMDB:DISPlay:GRADMethod {EMPH8 | EMPH7 | BIN8 | BIN7}  
 WFMDB:DISPlay:GRADMethod?

**Related Commands** [WFMDB:DISPlay:EMPHasis](#)

- Arguments**
- **EMPH8** selects a curve-driven grading method that utilizes eight display colors/intensities.  
 The curve is specified by the argument to the WFMDB:DISPlay:EMPHasis command, where the lower values emphasize bins with low counts and the higher values emphasize bins with high counts (range is 0 to 100).
  - **EMPH7** selects a curve-driven grading method that utilizes seven display colors/intensities.  
 The curve is specified by the argument to the WFMDB:DISPlay:EMPHasis command, where the lower values emphasize bins with low counts and the higher values emphasize bins with high counts (range is 0 to 100).
  - **BIN8** selects a binary grading method that uses eight display colors/intensities. This method assigns ranges of counts to colors/intensities by successively halving the maximum bin count and assigning the resulting ranges in brightest-to-darkest color/intensity order. If the maximum bin count is less than the number of display colors, a one-for-one mapping of counts to colors/intensities is used. BIN8 is the default setting.
  - **BIN7** selects a binary grading method that uses seven display colors/intensities. This method assigns ranges of counts to colors/intensities by successively halving the maximum bin count and assigning the resulting ranges in brightest-to-darkest color/intensity order. If the maximum bin count is less than the number of display colors, a one-for-one mapping of counts to colors/intensities is used.

**Examples** WFMDB:DISPLAY:GRADMETHOD EMPH7 sets the grading method for the all waveform databases to EMPH7, which emphasizes high or low bin counts,

depending on the emphasis value set using the `WFMDB:DISplay:EMPHasis` command.

`WFMDB:DISPLAY:GRADMETHOD?` might return `BIN8`, indicating that the grading method for the display of all waveform databases is set to binary 8.

## WFMDB:DISPlay:INVert

This command sets or queries whether the Color or Intensity grading is displayed inverted. When Invert is enabled, the mapping function that applies colors and intensity to the WfmDB significantly emphasizes the least occurring pixels. This is the equivalent to selecting or deselecting Invert Color/Intensity in the Display Options section of the WfmDB Setup dialog box.

**Group** Waveform Database

**Syntax** `WFMDB:DISPlay:INVert { ON | OFF | NR1 }`  
`WFMDB:DISPlay:INVert?`

**Related Commands** [WFMDB:DISPlay:EMPHasis](#), [WFMDB:DISPlay:GRADing](#)

- Arguments**
- ON enables the invert function.
  - OFF disables the invert function.
  - NR1 set to 0 disables on the invert function; any other integer enables the invert function.

**Examples** `WFMDB:DISPLAY:INVERT ON` enables the invert function.  
`WFMDB:DISPLAY:INVERT?` might return `WFMDB:DISPLAY:INVERT 0`, indicating that the invert function is disabled.

---

**NOTE.** *This query always returns 0 or 1.*

---

## WFMDB:WFMDB<x>? (Query Only)

This query only command returns the waveform database parameters for the WfmDB specified by x, which can be 1 through 4.

**Group** Waveform Database

**Syntax**      WFMDB:WFMDB<x>?

**Examples**      WFMDB:WFMDB2? might return WFMDB:WFMDB2:SOURCE REF1,MAIN;DISPLAY 1;ENABLE 1;PERSISTENCE:MODE INFPERSIST;COUNT 523;SAMPLES 130750.

## WFMDB:WFMDB<x>:CLEAR (No Query Form)

This command (no query form) resets to zero (clears) the counts in the specified WfmDB. The WfmDB is specified by x, which can be 1 through 4. (Sending the ACQUIRE:DATA:CLEAR command has the same effect on the WfmDB.)

**Group**          Waveform Database

**Syntax**          WFMDB:WFMDB<x>:CLEAR

**Related Commands**    [ACQUIRE:DATA:CLEAR](#)

**Examples**          WFMDB:WFMDB1:CLEAR clears the counts in WfmDB 1 (Waveform Database1).

## WFMDB:WFMDB<x>:DISPLAY

This command displays the specified WfmDB waveform. The query form of this command returns whether or not the specified WfmDB waveform is displayed. The WfmDB is specified by x, which can be 1 through 4. This command is the equivalent to checking or unchecking the Display box in the Database section of the Wfm Database Setup dialog box.

**Group**          Waveform Database

**Syntax**          WFMDB:WFMDB<x>:DISPLAY { ON | OFF | NR1 }  
WFMDB:WFMDB<x>:DISPLAY?

- Arguments**
- ON enables the display of the specified WfmDB waveform.
  - OFF disables the display of the specified WfmDB waveform. This is the default setting.
  - NR1 set to 0 disables the display of the specified WfmDB waveform; any other integer value enables the display of the specified WfmDB waveform.

**Examples** `WFMDB:WFMDB1:DISPLAY ON` enables the waveform display for Database 1.

`WFMDB:WFMDB2:DISPLAY?` might return `WFMDB:WFMDB2:DISPLAY 0`, indicating that the Database 2 waveform display is disabled.

---

**NOTE.** *This query always returns a 0 or 1.*

---

## WFMDB:WFMDB<x>:ENABLE

This command sets or queries whether the specified waveform database (WfmDB) is enabled. The WfmDB is specified by x, which can be 1 through 4. This command is the equivalent to selecting or deselecting the On check box in the Database section of the Wfm Database Setup dialog box. Use the `WFMDB:WFMDB<x>:DISPlay` command to control whether or not the waveform database is displayed.

**Group** Waveform Database

**Syntax** `WFMDB:WFMDB<x>:ENABLe { ON | OFF | NR1 }`  
`WFMDB:WFMDB<x>:ENABLe?`

**Related Commands** [WFMDB:WFMDB<x>:DISPlay](#)

- Arguments**
- ON enables the display of the specified WfmDB waveform.
  - OFF disables the display of the specified WfmDB waveform. This is the default setting.
  - NR1 set to 0 disables the display of the specified WfmDB waveform; any other integer value enables the display of the specified WfmDB waveform.

**Examples** `WFMDB:WFMDB1:ENABLE ON` enables the waveform display for Database 1.

`WFMDB:WFMDB2:ENABLE?` might return `WFMDB:WFMDB2:ENABLE 0`, indicating that the Database 2 waveform display is disabled.

---

**NOTE.** *This query always returns a 0 or 1.*

---

## WFMDB:WFMDB<x>:PERSistence:COUNT

This command sets or queries the Variable-persistence waveform count for the specified WfmDB. The WfmDB is specified by x, which can be 1 through 4. The

count applies to the Variable persistence mode, which can be selected with the command `WFMDB:WFMDB<x>:PERSistence:MODE VARPersist`.

Use of this command is equivalent to entering a value in the Waveforms Edit Box in the Persistence section of the Wfm Database Setup dialog box.

**Group** Waveform Database

**Syntax** `WFMDB:WFMDB<x>:PERSistence:COUNT <nr1>`  
`WFMDB:WFMDB<x>:PERSistence:COUNT?`

**Related Commands** [WFMDB:WFMDB<x>:PERSistence:MODE](#), [WFMDB:WFMDB<x>:PERSistence:SAMPLES?](#)

**Arguments** `<NR1>` (Range: 1 - 2000) sets the waveform count used for acquiring databases set to a persistence mode. The oldest waveforms accumulated in the database are removed as the waveforms accumulate beyond the specified count.

The maximum number of samples that can be accumulated in a variable persistence waveform database is 2,000,000 points. Depending on the record length of the timebase being acquired, the Persistence Count will be constrained so as not to exceed this maximum. For example, if the record length of the acquired timebase is 16,000 points, the Persistence count can take a maximum value of 125 (2,000,000 / 16,000).

**Examples** `WFMDB:WFMDB3:PERSISTENCE:COUNT 2E3` sets variable-persistence count for waveform database 3 to 2000 waveforms.

`WFMDB:WFMDB2:PERSISTENCE:COUNT?` might return 1000, indicating that the variable-persistence count for waveform database 2 is set to 1000 waveforms.

## WFMDB:WFMDB<x>:PERSistence:MODE

This command sets or queries which persistence mode, Infinite or Variable, that the specified WfmDB uses to accumulate waveform data. The WfmDB is specified by `x`, which can be 1 through 4. The modes are described under Arguments, below.

Use of this command is equivalent to selecting Infinite or Variable in the Persistence section of the Wfm Database Setup dialog box.

**Group** Waveform Database



**Syntax**      `WFMDB:WFMDB<x>:PERSistence:MODE {INFPersist | VARPersist}`  
`WFMDB:WFMDB<x>:PERSistence:MODE?`

**Related Commands**      [WFMDB:WFMDB<x>:PERSistence:SAMPles?](#), [WFMDB:WFMDB<x>:PERSistence:COUNT](#)

- Arguments**
- `INFPersist` sets `WfmDB<x>` to Infinite persistence mode, which continues displaying waveforms as they accumulate until the `WfmDB<x>` is cleared manually or by a control change (switching the database to Variable persistence, for example). `INFPERSIST` is the default setting.
  - `VARPersist` sets the waveform database to Variable persistence mode. Variable persistence keeps and displays accumulated data in the specified database until the user-specified waveform count is surpassed. Each waveform accumulated beyond the count removes the oldest waveform accumulated earlier in the database.

**Examples**      `WFMDB:WFMDB2:PERSISTENCE:MODE INFPERSIST` sets the persistence mode for waveform database 2 to Infinite.

`WFMDB:WFMDB3:PERSISTENCE:MODE?` might return `VARPERSIST`, indicating that the persistence mode for waveform database 3 is set to Variable.

## WFMDB:WFMDB<x>:PERSistence:SAMPles? (Query Only)

This query returns the sample count in effect for the Variable persistence mode of the specified `WfmDB`. The `WfmDB` is specified by `x`, which can be 1 through 4. The count applies to the Variable persistence mode which can be selected with the command `WFMDB:WFMDB<x>:PERSistence:MODE VARPersist`.

This query has no equivalent set command, since the number of samples cannot be set directly. The sample count returned is the product of the number of Waveforms specified (see the [WFMDB:WFMDB<x>:PERSistence:COUNT](#) command) and the current record length (see the [HORIZontal:MAIn:RECORDlength](#) command).

The count returned by this query can also be seen in the Samples readout in the Persistence section of the Wfm Database Setup dialog box.

**Group**      Waveform Database

**Syntax**      `WFMDB:WFMDB<x>:PERSistence:SAMPles?`

**Related Commands**      [WFMDB:WFMDB<x>:PERSistence:MODE](#), [WFMDB:WFMDB<x>:PERSistence:COUNT](#)

**Examples** `WFMDB:WFMDB<1>:PERSistence:SAMPles?` might return 50000, indicating that the oldest waveforms will be dropped from the database 1 as the sample count exceeds 50000 samples.

## WFMDB:WFMDB<x>:SOURce

This command sets or queries the source of the WfmDB specified by x, which can be 1 through 4. This command can optionally set the timebase. This is the equivalent to selecting a source (and timebase) in the Source control for the Database area of the WfmDB Setup dialog box. If you do not set the timebase, the default is Main.

**Group** Waveform Database

**Syntax** `WFMDB:WFMDB<x>:SOURce {CH<x> | MATH<x>} [, { MAIn | MAG1 | MAG2 }]`  
`WFMDB:WFMDB<x>:SOURce?`

- Arguments**
- `CH<x>` selects a channel waveform as the source for the specified WfmDB. The range for x is 1 through 8.
  - `MATH<x>` selects a math waveform as the source for the specified WfmDB. The range for x is 1 through 8.
  - `MAIn` (optional) selects Main as the timebase for the specified WfmDB display. This is the default timebase.
  - `MAG1` (optional) selects MAG1 as the timebase for the specified WfmDB display.
  - `MAG2` (optional) selects MAG2 as the timebase for the specified WfmDB display.

**Examples** `WFMDB:WFMDB1:SOURce CH4` sets Channel 4, Main timebase, as the source for WfmDb1.

`WFMDB:WFMDB2:SOURce?` might return `WFMDB:WFMDB2:SOURce MATH2,MAIn`, indicating that the Math 2 waveform is set as the source for WfmDB2 and is displayed with the Main timebase.

## WFMImpre? (Query Only)

This is a query only that returns the waveform formatting data (preamble) for the incoming waveform.

<b>Group</b>	Waveform Transfer
<b>Syntax</b>	WFMInpre?
<b>Examples</b>	WFMINPRE? might return WFMINPRE:ENCDG ASCII;BN_FMT RI;BYT_OR LSB;BYT_NR 4;BIT_NR 32;NR_PT 500;PT_FMT Y;XINCR 1.0000000000E-006;XZERO 16.0000000000E-009;XUNIT "";YMULT 0.10000000000;XMULT 100.0000000000E-012;XOFF 65.0000000000E-009;YZERO 1.0000000000E-006;YSCALE 5.0000000000E-003;YOFF -2;YUNIT "";WFMLABEL "".

## WFMInpre:BIT\_Nr? (Query Only)

This query only command returns the number of bits (width) per waveform point for the waveform to be transferred to the instrument.

---

**NOTE.** *This instrument supports only the 32-bit format.*

---

<b>Group</b>	Waveform Transfer
<b>Syntax</b>	WFMInpre:BIT_Nr?
<b>Related Commands</b>	<a href="#">WFMInpre:BYT_Nr?</a>
<b>Examples</b>	WFMINPRE:BIT_Nr? might return WFMINPRE:BIT_NR 32, indicating that there are 32 bits per waveform point.

## WFMInpre:BN\_Fmt

This command sets or queries the format of the binary or ascii data encoding of the waveform to be transferred to the instrument.

<b>Group</b>	Waveform Transfer
<b>Syntax</b>	WFMInpre:BN_Fmt {RI   FP} WFMInpre:BN_Fmt?
<b>Related Commands</b>	<a href="#">DATA:ENCdg</a> , <a href="#">WFMInpre:BYT_Or</a> , <a href="#">WFMInpre:ENCdg</a>

- Arguments**
- RI specifies signed integer data-point representation.
  - FP specifies floating-point data-point representation.

**Examples** `WFMINPRE:BN_FMT RI` specifies that the binary or ascii waveform data values are signed integer data points.

`WFMINPRE:BN_FMT?` might return `WFMPRE:BN_FMT FP`, indicating that the waveform data format for transferring a waveform is floating point.

## WFMinpre:BYT\_Nr? (Query Only)

This query only command returns the byte width of the waveform to be transferred to the instrument.

---

**NOTE.** *This instrument supports only a 4-byte width format.*

---

**Group** Waveform Transfer

**Syntax** `WFMinpre:BYT_Nr?`

**Related Commands** [WFMinpre:BIT\\_Nr?](#)

**Examples** `WFMINPRE:BYT_NR?` might return `WFMINPRE:BYT_NR 4`, indicating that the byte width for waveform data is four bits.

## WFMinpre:BYT\_Or

Selects which byte of binary waveform data is transmitted first during a waveform data transfer to the instrument.

**Group** Waveform Transfer

**Syntax** `WFMinpre:BYT_Or {LSB | MSB}`  
`WFMinpre:BYT_Or?`

**Related Commands** [DATA:ENCdg](#), [WFMinpre:BN\\_Fmt](#), [WFMinpre:ENCdg](#)

- Arguments**
- LSB selects the least-significant byte to be transmitted first.
  - MSB selects the most-significant byte to be transmitted first.
- Examples**
- `WFMINPRE:BYT_OR MSB` specifies that the most-significant byte in the waveform data will be transferred first.
- `WFMINPRE:BYT_OR?` might return `WFMINPRE:BYT_O MSB`, indicating that the most-significant byte in the waveform data will be transmitted first.

## WFMINpre:ENCdg

This command sets or queries the type of encoding used for waveform data transferred with the `CURVe` command.

**Group** Waveform Transfer

**Syntax** `WFMINpre:ENCdg {ASCIi | BINary}`  
`WFMINpre:ENCdg?`

**Related Commands** [DATA:ENCdg](#), [WFMINpre:BYT\\_Or](#), [WFMINpre:BN\\_Fmt](#), [CURVe](#)

- Arguments**
- `ASCIi` sets the encoding for waveform data transfer to ASCII format.
  - `BINary` sets the encoding for the waveform data transfer to binary format.
- Examples**
- `WFMINPRE:ENCDG ASCII` specifies that the input waveform data is in ASCII format.
- `WFMINPRE:ENCDG?` might return `WFMINPRE:ENCDG BIN`, indicating that the waveform data is in binary format.

## WFMINpre:NR\_Pt

This command sets or queries the record length of the input waveform. The query returns the allocated record length of the input waveform. This is the number of points in the curve plus any additional NULL values to fill the record. Any additional input points will be discarded. Valid record lengths are 50, 100, 250, 500, 1000, 2000, 4000, 8000, and 16000. When using this command to set the record length of the input waveform and the input value is different from one of the valid record lengths, the value will be coerced to the next higher valid record length. An event will then be generated.

<b>Group</b>	Waveform Transfer
<b>Syntax</b>	WFMInpre:NR_Pt <NR1> WFMInpre:NR_Pt?
<b>Related Commands</b>	<a href="#">CURVe</a> , <a href="#">DATA:STOP</a> , <a href="#">DATA:START</a> , <a href="#">DATA:DESTination</a>
<b>Arguments</b>	NR1  This argument is a valid record length. It can be 50, 100, 250, 500, 1000, 2000, 4000, 8000, and 16000.
<b>Examples</b>	WFMINPRE:NR_PT 2000 sets the record length to 2000.  WFMINPRE:NR_PT? might return WFMINPRE:NR_PT 4000, indicating that the record length for the input waveform is set to 4000.

## WFMInpre:PT\_Fmt

This command sets or queries the data point format of the incoming waveform.

<b>Group</b>	Waveform Transfer
<b>Syntax</b>	WFMInpre:PT_Fmt {Y   ENV}
<b>Related Commands</b>	<a href="#">DATA:DESTination</a>

- Arguments**
- Y specifies a standard waveform where one ASCII or binary point is transmitted for each point in the waveform record. Only y values are explicitly transmitted. Absolute coordinates are given by:
 
$$X(n) = 0 + (XINcr * n) + XZEro$$

$$Y(n) = (YSCALE * y(n)) + YZEro$$
 where  $n = 0,1,\dots$
  - ENV specifies that the waveform is transmitted as min and max pairs. Absolute coordinates are given by:
 
$$X(n) = XINcr (INT(n/2) + XZEro$$

$$Y(n(max)) = YSCALE * y(n(max)) + YZEro$$

$$Y(n(min)) = YSCALE * y(n(min)) + YZEro$$

**Examples**    `WFMINPRE:PT_FMT ENV` sets the waveform data point format to enveloped.  
`WFMINPRE:PT_FMT?` might return `WFMINPRE:PT_FMT Y`, which indicates that the data is transmitted as a standard waveform.

## WFMinpre:WFMLabel

This command sets or queries the label for the incoming waveform. Sending this command is equivalent to selecting Waveform Properties from the Utilities menu and entering a label in the Waveform Label field.

**Group**    Waveform Transfer

**Syntax**    `WFMinpre:WFMLabel <Qstring>`  
`WFMinpre:WFMLabel?`

**Arguments**    `Qstring` sets the label for the incoming waveform.

**Examples**    `WFMINPRE:WFMLABEL "PULSE TEST"` sets the label for the incoming waveform to Pulse Test.  
`WFMINPRE:WFMLABEL?` might return `WFMINPRE:WFMLABEL "TEST 4"`, indicating that the label for the incoming waveform is set to Test 4.

## WFMinpre:XINcr

This command sets or queries the interval between samples of the incoming waveform.

**Group** Waveform Transfer

**Syntax** WFMinpre:XINcr <NR3>  
WFMinpre:XINcr?

**Related Commands** [HORizontal:MAIn:RESolution?](#)

**Arguments** NR3 is the sampling interval in seconds per point.

**Examples** WFMINPRE:XINCR 1.0E-9 sets the horizontal sampling interval to 1 ns.  
WFMINPRE:XINCR? might return WFMINPRE:XINCR 5.0000000000E-009, indicating that the interval between samples is 5 ns.

## WFMinpre:XMUlt

This command sets or queries the horizontal scale factor (horizontal units per division) of the incoming waveform.

**Group** Waveform Transfer

**Syntax** WFMinpre:XMUlt <NR3>  
WFMinpre:XMUlt?

**Related Commands** [HORizontal:MAIn:SCAle](#)

**Arguments** NR3 is the horizontal scale factor of the incoming waveform.

**Examples** WFMINPRE:XMULT 1.0E-5 sets the horizontal scale factor to 10  $\mu$ s per division.  
WFMINPRE:XMULT? might return WFMINPRE:XMULT 1.0000000000E-009, indicating that the horizontal scale factor of the waveform to be transferred is set to 1 ns per division.



## WFMinpre:XOFF

This command sets or queries the horizontal position in horizontal units of the incoming waveform.

**Group** Waveform Transfer

**Syntax** WFMinpre:XOFF <NR3>  
WFMinpre:XOFF?

**Related Commands** [HORizontal:MAIn:POSition](#)

**Arguments** NR3 is the horizontal position in horizontal units.

**Examples** WFMinPRE:XOFF 6.5E-8 sets the horizontal position for the incoming waveform to 65 ns.

WFMinPRE:XOFF? might return WFMinPRE:XOFF 2.0E-8, indicating that the horizontal position for the waveform to be transferred is set to 20 ns.

## WFMinpre:XUNit

This command sets or queries the horizontal (x-axis) units of the incoming waveform.

**Group** Waveform Transfer

**Syntax** WFMinpre:XUNit <QString>  
WFMinpre:XUNit?

**Related Commands** [HORizontal:UNIts](#)

**Arguments** Qstring is a quoted string.

**Examples** WFMinPRE:XUNIT "S" sets the horizontal units to seconds.

WFMinPRE:XUNIT? might return WFMinPRE:XUNIT "S", indicating that the horizontal units of the incoming waveform are seconds.

## WFMinpre:XZero

This command sets or queries the horizontal (x-axis) origin offset (time of first point in this instrument) of the incoming waveform. The time of first point is the time interval between the trigger point and the first point in the record (waveform).

**Group** Waveform Transfer

**Syntax** WFMinpre:XZero <NR3>  
WFMinpre:XZero?

**Related Commands** [HORizontal:MAIn:TOFPoint?](#)

**Arguments** NR3 is the time interval between the trigger point and the first point in the waveform record.

**Examples** WFMINPRE:XZERO 20E-9 sets the horizontal origin offset (time of first point) to 20 ns for the incoming waveform. This is the time interval between the trigger point and the first point in the waveform record.

WFMINPRE:XZERO? might return WFMinpre:XZero 5.000000000000E-8, indicating that the time interval between the trigger point and the first point in the incoming waveform record is 50 ns.

## WFMinpre:YMUIt

This command sets or queries the vertical scale in vertical units per division of the incoming waveform returned by the previous CURVe? command.

**Group** Waveform Transfer

**Syntax** WFMinpre:YMUIt <NR3>  
WFMinpre:YMUIt?

**Related Commands** [CH<x>:SCAle](#), [CURVe](#)

**Arguments** NR3 is the vertical scale factor in YUNits (usually volts) per division of the incoming waveform.

- Examples** `WFMINPRE:YMULT 5.0E-3` sets the vertical scale factor of the incoming waveform to 5 mV per division.
- `WFMINPRE:YMULT?` might return `WFMINPRE:YMULT 0.100000000000`, indicating that the vertical scale of the incoming waveform is set to 100 mV per division.

## WFMINpre:YOff

This command sets or queries the vertical position of the incoming waveform in divisions.

**Group** Waveform Transfer

**Syntax** `WFMINpre:YOff <NR3>`  
`WFMINpre:YOff?`

**Related Commands** [CH<x>:POSition](#)

**Arguments** NR3 is the vertical position in divisions of the incoming waveform.

- Examples** `WFMINPRE:YOFF 1.0E+0` sets the vertical position to 1 division.
- `WFMINPRE:YOFF?` might return `WFMINPRE:YOFF -2` indicating that the vertical position of the incoming waveform is -2 divisions.

## WFMINpre:YSCALE

This command sets or queries the vertical unit scale factor of the incoming waveform. The value in vertical units is calculated by the following formula:  
*Vertical Units = (Raw Data) \* YSCALE + YZERO.*

**Group** Waveform Transfer

**Syntax** `WFMINpre:YSCALE <NR3>`  
`WFMINpre:YSCALE?`

**Arguments** NR3 is the value of the vertical scale factor. Note that the argument is always 1 when `DATA:SOURCE` is set to a math waveform.

**Examples** `WFMINPRE:YSCALE 9.20E-10` sets the vertical unit scale factor of the incoming waveform to 920E-12.

`WFMINPRE:YSCALE?` might return `WFMINPRE:YSCALE 9.20E10`, indicating that the vertical unit scale factor of the incoming waveform is 920 pV.

## WFMINpre:YUNit

This command sets or queries the vertical (y-axis) units of the incoming waveform.

**Group** Waveform Transfer

**Syntax** `WFMINpre:YUNit <QString>`  
`WFMINpre:YUNit?`

**Related Commands** [CH<x>:UNIts](#)

**Arguments** `QString` sets the vertical units of the incoming waveform.

**Examples** `WFMINPRE:YUNIT "w"` sets the vertical units of the incoming waveform to watts.

`WFMINPRE:YUNIT?` might return `WFMINPRE:YUNIT "V"`, indicating that the vertical units of the incoming waveform are volts.

## WFMINpre:YZEro

This command sets or queries the vertical offset voltage of the incoming waveform.

**Group** Waveform Transfer

**Syntax** `WFMINpre:YZEro <NR3>`  
`WFMINpre:YZEro?`

**Related Commands** [CH<x>:OFFSet](#)

**Arguments** `NR3` is the vertical offset in volts of the incoming waveform.

**Examples** `WFMINPRE:YZERO 5.0E-6` sets the vertical offset of the incoming waveform to 5  $\mu$ V

`WFMINPRE:YZERO?` might return `WFMINPRE:YZERO 1.0000000000E-006`, indicating that the vertical offset of the incoming waveform is 1  $\mu$ V.

## WFMOutpre? (Query Only)

This is a query only command that returns the preamble for the outgoing waveform. This data always corresponds to the previous `CURVe?` command (the returned parameters do not necessarily correspond to the current waveform settings made by the `DATA:SOURce` command). A `CURVe?` command must precede the `WFMOutpre?` query to get corresponding header/data information.

**Group** Waveform Transfer

**Syntax** `WFMOutpre?`

**Related Commands** [CURVe](#), [WAVFrm?](#)

**Examples** `WFMOUTPRE?` might return `WFMOUTPRE:ENCDG ASCII;BN_FMT RI;BYT_OR LSB;BYT_NR 4;BIT_NR 32;NR_PT 500;PT_FMT Y;XINCR 500.0000000000E-012;XZERO 18.00000000008E-009;XUNIT "";YMULT 0.10000000000;XMULT 500.0000000000E-012;XOFF 20.50000000008E-009;YZERO 0.00000000000;YSCALE 232.83064365432E-012;YOFF 0;YUNIT "";WFID "PULSE TEST, 100.0MV/DIV, 500.0PS/DIV, 500 POINTS, SAMPLE MODE";WFMLABEL "PULSE TEST".`

## WFMOutpre:BIT\_Nr? (Query Only)

This query only command returns the number of bits (width) per waveform point of the waveform to be transferred from the instrument.

---

**NOTE.** *This instrument supports only a 32-bit format.*

---

**Group** Waveform Transfer

**Syntax** `WFMOutpre:BIT_Nr?`

**Related Commands**    [WFMOutpre:BYT\\_Nr?](#)

**Examples**    WFMOUTPRE:BIT\_NR? might return WFMOUTPRE:BIT\_NR 32, indicating that there are 32 bits per waveform point.

## WFMOutpre:BN\_Fmt

This command sets or queries the format of the binary data encoding of the waveform to be transferred from the instrument. Sending this command will coerce DATA:ENCdg to a corresponding setting based on binary format and order.

**Group**    Waveform Transfer

**Syntax**    WFMOutpre:BN\_Fmt {RI | FP}  
WFMOutpre:BN\_Fmt?

**Related Commands**    [DATA:ENCdg](#), [WFMOutpre:BYT\\_Or](#), [WFMOutpre:ENCdg](#)

- Arguments**
- RI specifies signed integer data-point representation.
  - FP specifies floating-point data-point representation.

**Examples**    WFMOUTPRE:BN\_FMT RI specifies that the binary waveform data are signed integer data points.

WFMOUTPRE:BN\_FMT? might return WFMPRE:BN\_FMT FP, indicating that the current waveform data format is floating point binary.

## WFMOutpre:BYT\_Nr? (Query Only)

This query only command returns the byte width for the waveform to be transferred from the instrument.

---

**NOTE.** *This instrument supports only 4-byte data widths.*

---

**Group**    Waveform Transfer

**Syntax**    WFMOutpre:BYT\_Nr?

**Related Commands** [WFMOutpre:BIT\\_Nr?](#)

**Examples** `WFMOUTPRE:BYT_NR?` might return `WFMOUTPRE:BYT_NR 4`, indicating that the byte width for waveform data is 4.

## WFMOutpre:BYT\_Or

Selects which byte of binary waveform data is transmitted first during a waveform data transfer. If binary encoding is selected, this command will coerce `DATA:ENCdg` to corresponding settings based on the binary format and order.

**Group** Waveform Transfer

**Syntax** `WFMOutpre:BYT_Or {LSB | MSB}`  
`WFMOutpre:BYT_Or?`

**Related Commands** [DATA:ENCdg](#), [WFMOutpre:BN\\_Fmt](#), [WFMOutpre:ENCdg](#)

- Arguments**
- LSB selects the least-significant byte to be transmitted first.
  - MSB selects the most-significant byte to be transmitted first.

**Examples** `WFMOUTPRE:BYT_OR MSB` specifies that the most-significant byte in the waveform data will be transferred first.

`WFMOUTPRE:BYT_OR?` might return `WFMOUTPRE:BYT_OR MSB`, indicating that the most-significant byte in the waveform data will be transmitted first.

## WFMOutpre:ENCdg

This command sets or queries the type of encoding used for waveform data transferred with the `CURVe?` command. Using this command to change the type of encoding also changes the encoding set by the `DATA:ENCdg` command.

**Group** Waveform Transfer

**Syntax** `WFMOutpre:ENCdg {ASCIi | BINary}`  
`WFMOutpre:ENCdg?`

**Related Commands** [DATA:ENCdg](#), [WFMOutpre:BYT\\_Or](#), [WFMOutpre:BN\\_Fmt](#), [CURVe](#)

- Arguments**
- **ASCIi** sets the encoding for waveform data transfer to ASCII curve data. The output will be formatted to floating point or integer, based on how the data is stored in the instrument.
  - **BINary** sets the encoding for the waveform data transfer to binary format. This will coerce `DATA:ENCdg` to a corresponding setting based on binary format and order.

**Examples** `WFMOUTPRE:ENCdg ASCII` specifies that the waveform data is in ASCII format.

`WFMOUTPRE:ENCdg?` might return `WFMOUTPRE:ENCdg BINARY`, indicating that the waveform data is in binary format.

## WFMOutpre:NR\_Pt? (Query Only)

This is a query only command that returns the number of points that were returned by the previous `CURVe?` command. The number of points is the greater of the absolute value of `(DATA:STOP - DATA:START) + 1`, or the record length of the source waveform specified by the `DATA:SOUrce` command.

**Group** Waveform Transfer

**Syntax** `WFMOutpre:NR_Pt?`

**Related Commands** [CURVe](#), [DATA:STOP](#), [DATA:START](#), [DATA:SOUrce](#), [DATA:TIMEbase](#)

**Examples** `WFMOUTPRE:NR_PT?` might return `WFMOUTPRE:NR_PT 500`.

## WFMOutpre:PT\_Fmt? (Query Only)

This is a query only command that returns the data point format of the waveform returned by the previous `CURVe?` command.

**Group** Waveform Transfer

**Syntax** `WFMOutpre:PT_Fmt?`

**Related Commands** [ACQuire:MODE](#), [CURVe](#)



**Returns** This query returns either Y, which specifies a standard waveform where one ASCII or binary point is transmitted for each point in the waveform record, or ENV, which specifies that the waveform is transmitted as min and max pairs.

**Examples** WFMOUTPRE:PT\_FMT? might return WFMOUTPRE:PT\_FMT Y, which indicates that the data is transmitted as a standard waveform.

### WFMOutpre:WFId? (Query Only)

This is a query only command that returns a descriptive string of the waveform returned by the previous CURVe? command.

**Group** Waveform Transfer

**Syntax** WFMOutpre:WFId?

**Examples** WFMOUTPRE:WFID? might return WFMOUTPRE:WFID "C3, 100.0MV/DIV, 5.000NS, 500 POINTS, Sample mode".

### WFMOutpre:WFMLabel? (Query Only)

This is query only command that returns the label for the previous waveform transferred by the CURVe? command.

**Group** Waveform Transfer

**Syntax** WFMOutpre:WFMLabel?

**Examples** WFMOUTPRE:WFMLABEL? might return WFMOUTPRE:WFMLABEL "PULSE TEST".

### WFMOutpre:XINcr? (Query Only)

This is a query only command that returns the interval between samples of the waveform transferred by the CURVe? command.

**Group** Waveform Transfer

**Syntax** WFMOutpre:XINcr?

**Related Commands** [HORizontal:MAIn:RESolution?](#)

**Examples** WFMOUTPRE:XINCR? might return WFMOUTPRE:XINCR 5.0000000000E-009, indicating that the interval between samples of the previously transferred waveform was 5 ns.

## WFMOutpre:XMUlt? (Query Only)

This is a query only command that returns the horizontal scale factor of the waveform returned by the previous CURVe? command.

**Group** Waveform Transfer

**Syntax** WFMOutpre:XMUlt?

**Related Commands** [HORizontal:MAIn:SCAle](#)

**Examples** WFMOUTPRE:XMULT? might return WFMOUTPRE:XMULT 5.0000000000E-006, indicating that the horizontal scale factor of the previously transferred waveform was set to 5  $\mu$ s per division.

## WFMOutpre:XOff? (Query Only)

This is a query only command that returns the horizontal position in horizontal units of the returned waveform.

**Group** Waveform Transfer

**Syntax** WFMOutpre:XOff?

**Related Commands** [HORizontal:MAIn:POSition](#)

**Examples** WFMOUTPRE:XOFF? might return WFMOUTPRE:XOFF 20.5000000008E-009, indicating that the horizontal position for the previously transferred waveform was set to 20.5 ns.

## WFMOutpre:XUNit? (Query Only)

This is a query only command that returns the horizontal units of the waveform returned by the previous CURVe? command.

**Group** Waveform Transfer

**Syntax** WFMOutpre:XUNit?

**Related Commands** [HORizontal:UNIts](#)

**Examples** WFMOUTPRE:XUNIT? might return WFMOUTPRE:XUNIT "S", indicating that the horizontal units of the previously acquired waveform were seconds.

## WFMOutpre:XZEro? (Query Only)

This is a query only command that returns the time of first point (sample) of the waveform returned by the previous CURVe? command.

**Group** Waveform Transfer

**Syntax** WFMOutpre:XZEro?

**Related Commands** [HORizontal:MAIn:TOFPoint?](#)

**Examples** WFMOUTPRE:XZERO? might return WFMOUTPRE:XZERO 19.00000000008E-009, indicating that the time interval between the trigger point and the first point in the previous waveform record was 19 ns.

## WFMOutpre:YMUIt? (Query Only)

This is a query only command that returns the vertical scale factor in vertical units per division of the waveform returned by the previous CURVe? command.

**Group** Waveform Transfer

**Syntax** WFMOutpre:YMUIt?

**Related Commands**    [CH<x>:SCALE](#)

**Examples**    WFMOUTPRE:YMULT? might return WFMOUTPRE:YMULT 0.100000000000, indicating that the vertical scale of the previously transferred waveform was set to 100 mV per division.

## WFMOutpre:YOFF? (Query Only)

This is a query only command that returns the vertical position of the waveform returned by the previous CURVe? command.

**Group**    Waveform Transfer

**Syntax**    WFMOutpre:YOFF?

**Related Commands**    [CH<x>:POSITION](#)

**Examples**    WFMOUTPRE:YOFF? might return WFMOUTPRE:YOFF 0, indicating that the vertical position of the previously transferred waveform was at the centerline.

## WFMOutpre:YSCALE? (Query Only)

This query only command returns vertical unit scale factor of the incoming waveform by the previous CURVe? command. The value in vertical units is calculated by the following formula: *Vertical Units = (Raw Data) x YSCALE + YZERO*.

---

**NOTE.** *If the DATA:SOURCE is a math waveform, the WFMOUTPRE:YSCALE value is always 1. If the data source is a reference waveform, then the WFMOUTPRE:YSCALE value depends on the original source data.*

---

**Group**    Waveform Transfer

**Syntax**    WFMOutpre:YSCALE?

**Examples**    WFMOUTPRE:YSCALE? might return WFMOUTPRE:YSCALE 9.200000000000E10, indicating that the vertical unit scale factor of the incoming waveform is 920p.

## WFMOutpre:YUNit? (Query Only)

This is a query only command that returns the vertical units of the waveform returned by the previous CURVe? command.

**Group** Waveform Transfer

**Syntax** WFMOutpre:YUNit?

**Related Commands** [CH<x>:UNIts](#)

**Examples** WFMOUTPRE:YUNIT? might return WFMOUTPRE:YUNIT "V", indicating that the vertical units of the previously transferred waveform were volts.

## WFMOutpre:YZero? (Query Only)

This is a query only command that returns the vertical offset of the waveform returned by the previous CURVe? command.

**Group** Waveform Transfer

**Syntax** WFMOutpre:YZero?

**Related Commands** [CH<x>:OFFSet](#)

**Examples** WFMOUTPRE:YZERO? might return WFMOUTPRE:YZERO 1.0000000000E-001, indicating that the vertical offset of the previously transferred waveform was 100 mV.



---

# Status and Events

The instrument provides a status and event reporting system for the GPIB interfaces. This system informs you of certain significant events that occur within the instrument.

The instrument status handling system consists of five 8-bit registers and two queues for each interface. The remaining Status subtopics describe these registers and components. They also explain how the event handling system operates.

## Registers

The registers in the event handling system fall into two functional groups:

- Status Registers contain information about the status of the instrument. They include the Standard Event Status Register (SESR).
- Enable Registers determine whether selected types of events are reported to the Status Registers and the Event Queue. They include the Device Event Status Enable Register (DESER), the Event Status Enable Register (ESER), and the Service Request Enable Register (SRER).

**Status** The Standard Event Status Register (SESR) and the Status Byte Register (SBR) record certain types of events that may occur while the instrument is in use. IEEE Std 488.2-1987 defines these registers.

Each bit in a Status Register records a particular type of event, such as an execution error or message available. When an event of a given type occurs, the instrument sets the bit that represents that type of event to a value of one. (You can disable bits so that they ignore events and remain at zero. (See Registers). Reading the status registers tells you what types of events have occurred.

**The Standard Event Status Register (SESR).** The SESR records eight types of events that can occur within the instrument. Use the \*ESR? query to read the SESR register. Reading the register clears the bits of the register so that the register can accumulate information about new events.

---

**NOTE.** TekVISA applications use SESR bit 6 to respond to any of several events, including some front panel actions.

---

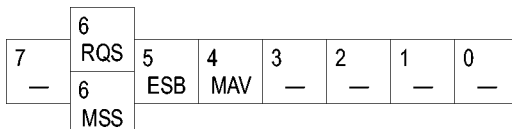
7	6	5	4	3	2	1	0
PON	URQ	CME	EXE	DDE	QYE	RQC	OPC

**Table 3-1: SESR Bit Functions**

Bit	Function	
7 (MSB)	PON	Power On. Shows that the oscilloscope was powered on. On completion, the diagnostic tests also set this bit.
6	URQ	User Request. Indicates that an application event has occurred. *See preceding note.
5	CME	Command Error. Shows that an error occurred while the oscilloscope was parsing a command or query.
4	EXE	Execution Error. Shows that an error executing a command or query.
3	DDE	Device Error. Shows that a device error occurred.
2	QYE	Query Error. Either an attempt was made to read the Output Queue when no data was present or pending, or that data in the Output Queue was lost.
1	RQC	Request Control. This is not used.
0 (LSB)	OPC	Operation Complete. Shows that the operation is complete. This bit is set when all pending operations complete following an *OPC command.

**The Status Byte Register (SBR).** Records whether output is available in the Output Queue, whether the instrument requests service, and whether the SESR has recorded any events.

Use a Serial Poll or the \*STB? query to read the contents of the SBR. The bits in the SBR are set and cleared depending on the contents of the SESR, the Event Status Enable Register (ESER), and the Output Queue. When you use a Serial Poll to obtain the SBR, bit 6 is the RQS bit. When you use the \*STB? query to obtain the SBR, bit 6 is the MSS bit. Reading the SBR does not clear the bits.



**Table 3-2: SBR Bit Functions**

Bit	Function	
7 (MSB)	—	Not used.
6	RQS	Request Service. Obtained from a serial poll. Shows that the oscilloscope requests service from the GPIB controller.
6	MSS	Master Status Summary. Obtained from *STB? query. Summarizes the ESB and MAV bits in the SBR.
5	ESB	Event Status Bit. Shows that status is enabled and present in the SESR.
4	MAV	Message Available. Shows that output is available in the Output Queue.



**Table 3-2: SBR Bit Functions (cont.)**

Bit	Function
3	Not used.
2	Not used.
1-0	Not used.

**Enable**

DESER, ESER, and SRER allow you to select which events are reported to the Status Registers and the Event Queue. Each Enable Register acts as a filter to a Status Register (the DESER also acts as a filter to the Event Queue) and can prevent information from being recorded in the register or queue.

Each bit in an Enable Register corresponds to a bit in the Status Register it controls. In order for an event to be reported to a bit in the Status Register, the corresponding bit in the Enable Register must be set to one. If the bit in the Enable Register is set to zero, the event is not recorded.

Various commands set the bits in the Enable Registers. The Enable Registers and the commands used to set them are described below.

**The Device Event Status Enable Register (DESER).** This register controls which types of events are reported to the SESR and the Event Queue. The bits in the DESER correspond to those in the SESR.

Use the DESE command to enable and disable the bits in the DESER. Use the DESE? query to read the DESER.

7	6	5	4	3	2	1	0
PON	URQ	CME	EXE	DDE	QYE	RQC	OPC

**The Event Status Enable Register (ESER).** This register controls which types of events are summarized by the Event Status Bit (ESB) in the SBR. Use the \*ESE command to set the bits in the ESER. Use the \*ESE? query to read it.

7	6	5	4	3	2	1	0
PON	URQ	CME	EXE	DDE	QYE	RQC	OPC

**The Service Request Enable Register (SRER).** This register controls which bits in the SBR generate a Service Request and are summarized by the Master Status Summary (MSS) bit.

Use the \*SRE command to set the SRER. Use the \*SRE? query to read the register. The RQS bit remains set to one until either the Status Byte Register is read with a Serial Poll or the MSS bit changes back to a zero.

7	6	5	4	3	2	1	0
—	—	ESB	MAV	—	—	—	—

**\*PSC Command** The \*PSC command controls the Enable Registers contents at power-on. Sending \*PSC 1 sets the Enable Registers at power on as follows:

- DESER 255 (equivalent to a DESe 255 command)
- ESER 0 (equivalent to an \*ESE 0 command)
- SRER 0 (equivalent to an \*SRE 0 command)

Sending \*PSC 0 lets the Enable Registers maintain their values in nonvolatile memory through a power cycle.

---

**NOTE.** *To enable the PON (Power On) event to generate a Service Request, send \*PSC 0, use the DESe and \*ESE commands to enable PON in the DESER and ESER, and use the \*SRE command to enable bit 5 in the SRER. Subsequent power-on cycles will generate a Service Request.*

---

## Queues

The \*PSC command controls the Enable Registers contents at power-on. Sending \*PSC 1 sets the Enable Registers at power on as follows:

**Output Queue** The instrument stores query responses in the Output Queue and empties this queue each time it receives a new command or query message after an <EOM>. The controller must read a query response before it sends the next command (or query) or it will lose responses to earlier queries.




---

**CAUTION.** *When a controller sends a query, an <EOM>, and a second query, the instrument normally clears the first response and outputs the second while reporting a Query Error (QYE bit in the ESER) to indicate the lost response. A fast controller, however, may receive a part or all of the first response as well. To avoid this situation, the controller should always read the response immediately after sending any terminated query message or send a DCL (Device Clear) before sending the second query.*

---

**Event Queue** The Event Queue stores detailed information on up to 33 events. If more than 32 events stack up in the Event Queue, the 32nd event is replaced by event code 350, "Queue Overflow."

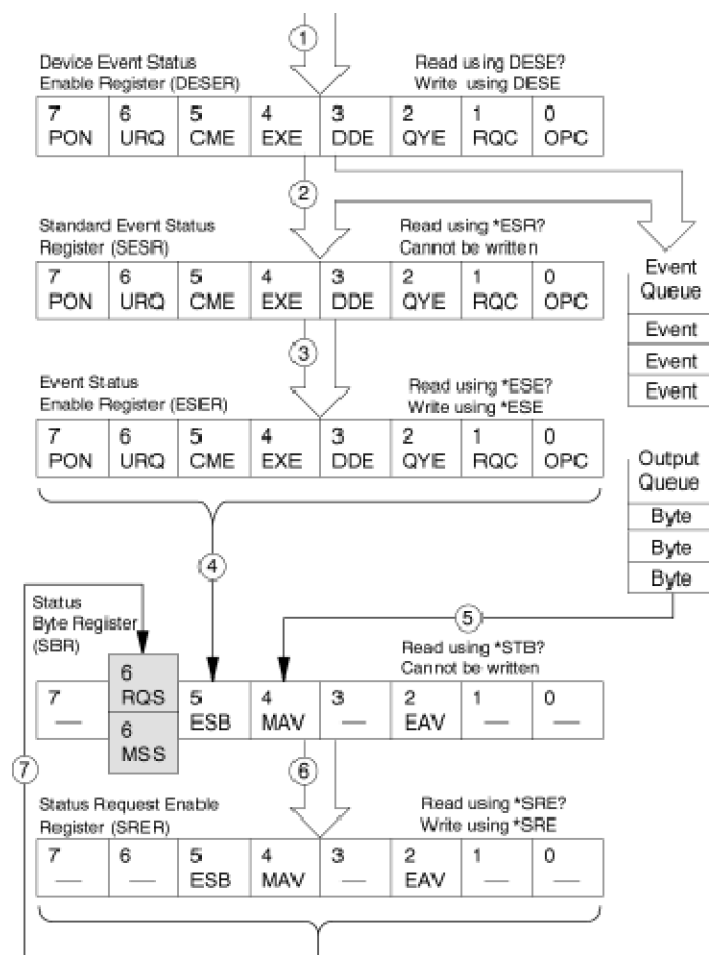
Read the Event Queue with the EVENT? query (which returns only the event number), with the EVMSG? query (which returns the event number and a text description of the event), or with the ALLEV? query (which returns all the event numbers along with a description of the event). Reading an event removes it from the queue.

Before reading an event from the Event Queue, you must use the \*ESR? query to read the summary of the event from the SESR. This makes the events summarized by the \*ESR? read available to the EVENT? and EVMSG? queries, and empties the SESR.

Reading the SESR erases any events that were summarized by previous \*ESR? reads but not read from the Event Queue. Events that follow an \*ESR? read are put in the Event Queue but are not available until \*ESR? is used again.

## Event Handling Sequence

The figure below shows how to use the status and event handling system. In the explanation that follows, numbers in parentheses refer to numbers in the figure.



When an event occurs, a signal is sent to the DESER (1). If that type of event is enabled in the DESER (that is, if the bit for that event type is set to 1), the appropriate bit in the SESR is set to one, and the event is recorded in the Event

Queue (2). If the corresponding bit in the ESER is also enabled (3), then the ESB bit in the SBR is set to one (4).

When output is sent to the Output Queue, the MAV bit in the SBR is set to one (5).

When a bit in the SBR is set to one and the corresponding bit in the SRER is enabled (6), the MSS bit in the SBR is set to one and a service request is generated (7). Bit 2 of the SBR (MAV) flags the presence of events in the queue.

## Synchronization Methods

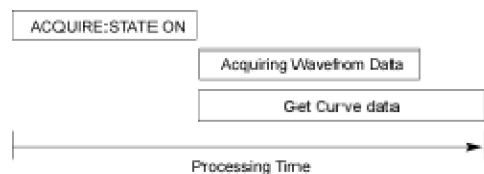
Although most GPIB commands are completed almost immediately after being received by the instrument, some commands start a process that requires more time. For example, once a single sequence acquisition command is executed, depending upon the applied signals and trigger settings, it may be a few seconds before the acquisition is complete. Rather than remain idle while the operation is in process, the instrument will continue processing other commands. This means that some operations will not be completed in the order that they were sent.

Sometimes the result of an operation depends on the result of an earlier operation. A first operation must complete before the next one gets processed. The instrument status and event reporting system provides ways to do this.

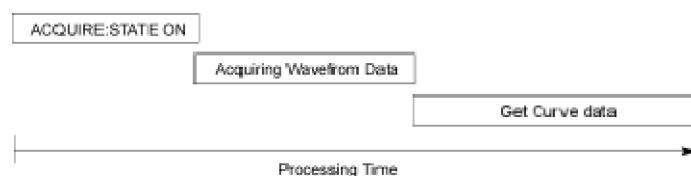
For example, synchronization may be used to ensure that the curve? query command returns waveform data that is consistent with the current instrument settings. You could use the following sequence to do this:

```
/** Sets up conditional acquisition **/  
ACQUIRE:STATE OFF  
SELECT:CH1 ON  
HORIZONTAL:RECORDLENGTH 500  
ACQUIRE:MODE SAMPLE  
ACQUIRE:STOPAFTER COUNT 1  
ACQUIRE:STOPAFTER:MODE CONDITION  
/** Sets up the data preamble **/  
DATA:START 1  
DATA:STOP 500  
DATA:ENCDG RIBINARY  
/** Clear data and acquire waveforms until conditional  
stop occurs**/  
ACQUIRE:DATA:CLEAR  
ACQUIRE:STATE ON  
/** Synchronize the operations by using *WAI, Busy?,  
*OPC, or *OPC?/  
<Synchronization command>  
/** Get the curve data **/  
CURVE?
```

The acquisition of the waveform requires extended processing time. It may not finish before the instrument executes the CURVE? query (see the following figure). This can result in incorrect curve values.



To ensure that the instrument completes waveform acquisition before attempting to execute the CURVE? query, you can synchronize the program. The figure below shows the desired processing sequence.



You can use four commands to synchronize the operation of the instrument with your application program: \*WAI, BUSY?, \*OPC, and \*OPC?

**\*WAI** You can force commands to execute sequentially by using the \*WAI command. This command forces completion of the previous commands before processing new ones.

The same command sequence using the \*WAI command for synchronization looks like this:

```
/** Sets up conditional acquisition **/
ACQUIRE:STATE OFF
SELECT:CH1 ON
HORIZONTAL:RECORDLENGTH 500
ACQUIRE:MODE SAMPLE
ACQUIRE:STOPAFTER COUNT 1
ACQUIRE:STOPAFTER:MODE CONDITION
/** Sets up the data preamble **/
DATA:START 1
DATA:STOP 500
DATA:ENCDG RIBINARY
/** Clear data and then acquire waveforms until
conditional stop occurs**/
ACQUIRE:DATA:CLEAR
ACQUIRE:STATE ON
/** wait until the acquisition is complete before
querying the curve data**/
*WAI
/** Get the curve data **/
```

### CURVE?

Although \*WAI is one of the easiest ways to achieve synchronization, it is also the most costly. The processing time of the instrument is slowed since it is processing a single command at a time. This time could be spent doing other tasks.

The controller can continue to write commands to the input buffer of the instrument, but the commands will not be processed by the instrument until all operations in process are complete. If the input buffer becomes full, the controller will be unable to write more commands to the buffer. This can cause a timeout.

**BUSY** The BUSY? query allows you to find out whether the instrument is busy processing a command that has an extended processing time such as single-sequence acquisition.

The same command sequence, using the BUSY? query for synchronization, looks like this:

```

/** Sets up conditional acquisition **/
ACQUIRE:STATE OFF
SELECT:CH1 ON
HORIZONTAL:RECORDLENGTH 500
ACQUIRE:MODE SAMPLE
ACQUIRE:STOPAFTER COUNT 1
ACQUIRE:STOPAFTER:MODE CONDITION
/** Sets up the data preamble **/
DATA:START 1
DATA:STOP 500
DATA:ENCDG RIBINARY
/** Clear data and then acquire waveforms until
conditional stop occurs**/
ACQUIRE:DATA:CLEAR
ACQUIRE:STATE ON
/** wait until the acquisition is complete before
querying the curve data**/
while BUSY? keep looping
/** Get the curve data **/
CURVE?
    
```

This sequence lets you create your own wait loop rather than using the \*WAI command. The BUSY? query helps you avoid time-outs caused by writing too many commands to the input buffer. The controller is still tied up, though, and the repeated BUSY? query will result in more bus traffic.

**\*OPC** If the corresponding status registers are enabled, the \*OPC command sets the OPC bit in the Standard Event Status Register (SESR) when an operation is complete. You achieve synchronization by using this command with either a serial poll or service request handler.

**Serial Poll Method:** Enable the OPC bit in the Device Event Status Enable Register (DESER) and the Event Status Enable Register (ESER) using the DESE and \*ESE commands.

When the operation is complete, the OPC bit in the Standard Event Status Register (SESR) will be enabled and the Event Status Bit (ESB) in the Status Byte Register will be enabled.

The same command sequence using the \*OPC command for synchronization with serial polling looks like this:

```

/** Sets up conditional acquisition **/
ACQUIRE:STATE OFF
SELECT:CH1 ON
HORIZONTAL:RECORDLENGTH 500
ACQUIRE:MODE SAMPLE
ACQUIRE:STOPAFTER COUNT 1
ACQUIRE:STOPAFTER:MODE CONDITION
/** Enable the status registers **/
DESE 1
*ESE 1
*SRE 0
/** Sets up the data preamble **/
DATA:START 1
DATA:STOP 500
DATA:ENCDG RIBINARY
/** Clear data and then acquire waveforms until
conditional stop occurs**/
ACQUIRE:DATA:CLEAR
ACQUIRE:STATE ON
/** wait until the acquisition is complete before
querying the curve data**/
*OPC
while serial poll = 0, keep looping
/** Get the curve data **/
CURVE?

```

This technique requires less bus traffic than did looping on BUSY.

**Service Request Method:** Enable the OPC bit in the Device Event Status Enable Register (DESER) and the Event Status Enable Register (ESER) using the DESE and \*ESE commands.

You can also enable service requests by setting the ESB bit in the Service Request Enable Register (SRER) using the \*SRE command. When the operation is complete, a Service Request will be generated.

The same command sequence using the \*OPC command for synchronization looks like this:

```

/** Sets up conditional acquisition **/

```

```

ACQUIRE:STATE OFF
SELECT:CH1 ON
HORIZONTAL:RECORDLENGTH 500
ACQUIRE:MODE SAMPLE
ACQUIRE:STOPAFTER:CONDITION ACQWFMS
ACQUIRE:STOPAFTER:COUNT 100
ACQUIRE:STOPAFTER:MODE CONDITION
/** Enable the status registers **/
DESE 1
*ESE 1
*SRE 32
/** Set up measurement parameters **/
MEASUREMENT:MEAS1:TYPE AMPLITUDE
MEASUREMENT:MEAS1:SOURCE CH1
/** Acquire waveforms until conditional stop occurs**/
ACQUIRE:DATA:CLEAR
ACQUIRE:STATE ON
/**wait until the acquisition is complete
before taking the measurement**/
*OPC
/**The program can now do different tasks such as
talk to other devices. The SRQ, when it comes,
interrupts those tasks and returns control to this
task.**/
/** Take amplitude measurement **/
MEASUREMENT:MEAS1:VALUE?

```

This technique is more efficient but requires more sophisticated programming.

**\*OPC?** The \*OPC? query places a 1 in the Output Queue once an operation is complete. A timeout could occur if you try to read the output queue before there is any data in it.

The same command sequence using the \*OPC? query for synchronization looks like this:

```

/** Sets up conditional acquisition **/
ACQUIRE:STATE OFF
SELECT:CH1 ON
HORIZONTAL:RECORDLENGTH 500
ACQUIRE:MODE SAMPLE
ACQUIRE:STOPAFTER:COUNT 1
ACQUIRE:STOPAFTER:MODE CONDITION
/** Set up measurement parameters **/
MEASUREMENT:MEAS1:TYPE AMPLITUDE
MEASUREMENT:MEAS1:SOURCE CH1
/** Clear data, and then acquire waveforms until
conditional stop occurs**/

```



```

ACQUIRE:DATA:CLEAR
ACQUIRE:STATE ON
/** wait until the acquisition is complete
before querying the curve data**/
*OPC?
/** wait for read from Output Queue **/
/** Get the curve data **/
CURVE?

```

This is the simplest approach. It requires no status handling or loops. However, you must set the controller timeout for a longer period of time than that used by the acquisition operation.

## Messages

This section covers all the programming interface event messages the instrument generates in response to commands and queries.

For most messages, a secondary message from the instrument gives more detail about the cause of the error or the meaning of the message. This message is part of the message string and is separated from the main message by a semicolon.

Each message is the result of an event. Each type of event sets a specific bit in the SESR and is controlled by the equivalent bit in the DESER. Thus, each message is associated with a specific SESR bit. In the message tables, the associated SESR bit is specified in the table title, with exceptions noted with the error message text.

**No Event** The following table shows the messages when the system has no events or status to report. These have no associated SESR bit.

**Table 3-3: No Event Messages**

Code	Message
0	No events to report; queue empty
1	No events to report; new events pending *ESR?

**Command Error** The following table shows the command error messages generated by improper syntax. Check that the command is properly formed and that it follows the rules in the section on Command Syntax.

**Table 3-4: Command Error Messages (CME Bit 5)**

Code	Message
100	Command error
101	Invalid character
102	Syntax error

**Table 3-4: Command Error Messages (CME Bit 5) (cont.)**

<b>Code</b>	<b>Message</b>
103	Invalid separator
104	Data type error
105	GET not allowed
108	Parameter not allowed
109	Missing parameter
110	Command header error
111	Header separator error
112	Program mnemonic too long
113	Undefined header
114	Header suffix out of range
118	Query not allowed
120	Numeric data error
121	Invalid character in number
123	Exponent too large
124	Too many digits
128	Numeric data not allowed
130	Suffix error
131	Invalid suffix
134	Suffix too long
138	Suffix not allowed
140	Character data error
141	Invalid character data
144	Character data too long
	Character data not allowed
150	String data error
151	Invalid string data
152	String data too long
158	String data not allowed
160	Block data error
161	Invalid block data
168	Block data not allowed
170	Expression error
171	Invalid expression
178	Expression data not allowed

**Execution Error**

The following table lists the execution errors that are detected during execution of a command.

**Table 3-5: Execution Error Messages (EXE Bit 4)**

<b>Code</b>	<b>Message</b>
200	Execution error
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	List not same length
230	Data corrupt or stale
231	Data questionable
240	Hardware error
241	Hardware missing
242	Hardware configuration error
243	Hardware I/O device error
244	Invalid printer selected
250	Mass storage error
251	Missing mass storage
252	Missing media
253	Corrupt media
254	Media full
255	Directory full
256	File name not found
257	File name error
258	Media protected
260	Expression error
261	Math error in expression
286	Program runtime error
2200	Measurement error, Measurement system error
2201	Measurement error, Zero period

Table 3-5: Execution Error Messages (EXE Bit 4) (cont.)

Code	Message
2202	Measurement error, No period found
2203	Measurement error, No period, second waveform
2204	Measurement error, Low signal amplitude
2205	Measurement error, Low amplitude
2206	Measurement error, Invalid gate
2207	Measurement error, Measurement overflow
2208	Measurement error, Waveform does not cross Mid Ref
2209	Measurement error, No second Mid Ref crossing
2210	Measurement error, No Mid Ref crossing, second waveform
2211	Measurement error, No backwards Mid Ref Crossing
2212	Measurement error, No negative crossing
2213	Measurement error, No positive crossing
2214	Measurement error, No crossing
2215	Measurement error, No crossing, second waveform
2216	Measurement error, No crossing, target waveform
2217	Measurement error, Constant waveform
2218	Measurement error, Unused
2219	Measurement error, No valid edge – No arm sample
2220	Measurement error, No valid edge – No arm cross
2221	Measurement error, No valid edge – No trigger cross
2222	Measurement error, No valid edge – No second cross
2223	Measurement error, Waveform mismatch
2224	Measurement error, WAIT calculating
2225	Measurement error, No waveform to measure
2226	Measurement error, Null Waveform
2227	Measurement error, Positive and Negative Clipping
2228	Measurement error, Positive Clipping
2229	Measurement error, Negative Clipping
2230	Measurement error, High Ref < Low Ref
2231	Measurement error, no statistics available
2235	Math error, Invalid math description
2236	Math error, Reference waveform is invalid
2237	Math error, Out of acquisition memory
2241	Waveform request is invalid
2243	This measurement cannot be performed on this type of waveform
2244	Source waveform is not active
2245	Saveref error, Selected channel is turned off

**Table 3-5: Execution Error Messages (EXE Bit 4) (cont.)**

<b>Code</b>	<b>Message</b>
2246	Saveref error, Selected channel data invalid
2248	This ref cannot be activated
2249	Reference deletion error, Waveform in use
2301	Cursor error, Off-screen
2303	Cursor error, Cursor source waveform is off
2304	Cursor error, Cursors are off
2321	Histogram warning, histogram turned off
2400	Not enough memory available
2401	This channel cannot be activated
2402	Math/Meas/Histo have circular definition
2410	Empty math string
2411	Syntax error in math string
2412	Semantic error in math string
2413	Math expression is too complex
2420	Histogram cannot be performed on this type of waveform
2425	Mask counting cannot be performed on this type of waveform
2430	WfmDB cannot be built on this type of waveform
2431	No Waveform Database resource available
2435	Selected Channel is not a TDR capable channel
2440	State Change – Reject Conditional Stop Action
2441	Selected Condition is Not Active
2450	TDR Invalid with FrameScan active
2451	Envelope Mode Invalid with FrameScan active
2455	RTL Hardware Warning
2456	RTL Software Error – severe problem
2457	RTL Software Warning
2458	Diagnostic Error
2459	Compensation Error
2500	Mask error – Mask Margin Boundary violated
2501	TDR coerced off – TDR operation not allowed with probe attached
2502	Waveform Database Source in use
2503	Clock Recovery not available
2504	Record length incompatible with scale
2505	Autoset Execute Warning – No waveforms enabled
2506	Error saving waveform to file
2507	FrameScan and Average Complete incompatible
2508	Selected filter is not available

**Table 3-5: Execution Error Messages (EXE Bit 4) (cont.)**

<b>Code</b>	<b>Message</b>
2509	Gated Trigger is not available
2510	Warning – measurement requires waveform database, and none is available
2511	Error saving setup – invalid directory path or file
2512	Could not close file
2513	Could not open file to read
2514	Can't write to file; media access violation
2515	Reference slot is out of range (1 – 8)
2516	Reference slot is active; cannot recall data into active slot
2517	Error reading waveform file
2518	Incompatible file version
2519	File path is not valid
2520	Waveform database is not properly released
2521	Reference data are not properly released
2522	Unable to restore setup
2523	Not enough mag points (<10) to save
2524	Directory does not exist; unable to create the directory
2525	Histogram contains invalid data
2526	The file can not be removed
2527	Source already active in another Database
2528	User-defined clock recovery not available
2529	TDR incompatible with Phase Correction Mode
2530	Phase Ref source not available
2531	TDR Step incompatible with phase Correction
2532	Phase Correction mode is off
2533	Clock Recovery source not available
2534	Clock Recovery rate not available
2535	Unit selection incompatible with Phase Ref mode
2536	Trigger Mode incompatible with Trigger Source
2537	Pattern Sync option not installed
2539	Pattern Sync Parameter not settable
2540	Autoset error, Failed to complete
2541	Autoset warning, Nothing to autoset
2542	Autoset error, Signal offset out of range
2543	Autoset error, Trigger not found
2544	Autoset error, Trigger Amplitude too small
2545	Autoset error, Signal Amplitude too small
2546	Autoset error, Signal Amplitude too large

**Table 3-5: Execution Error Messages (EXE Bit 4) (cont.)**

<b>Code</b>	<b>Message</b>
2547	Autoset error, Signal period not found
2548	Autoset error, Eye not found
2549	Autoset error, Phase Ref Clock too slow
2550	Application name error
2551	AutoSync error, No AutoSync on static wfm
2552	AutoSync error, No waveforms enabled
2553	AutoSync error, Mid Ref not found
2554	AutoSync error, Pattern Length not found
2555	AutoSync error, Data Rate not found
2556	AutoSync error, Trigger period not found
2557	AutoSync error, No AutoSync options selected
2558	AutoSync error, Trigger source is not PSync
2559	Autoset error, No TDR autoset on Ref wfm
2560	Autoset error, No TDR autoset on Math wfm
2561	Autoset error, Incident Edge not found
2562	Autoset error, Reflection not found
2563	Autoset error, Selected Channel is not a TDR capable channel

**Device Error**

The following table lists the device errors that can occur during instrument operation. These errors may indicate that the instrument needs repair.

**Table 3-6: Device Error Messages (DDE Bit 3)**

<b>Code</b>	<b>Message</b>
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
316	RTL Hardware Error, severe problem
350	Queue overflow (does not set DDE bit)

**System Event**

The following table lists the system event messages. These messages are generated whenever certain system conditions occur.

**Table 3-7: System Event Messages**

<b>Code</b>	<b>Message</b>
400	Query event
401	Power on (PON bit 7 set)
402	Operation complete (OPC bit 0 set)
403	User request (URQ bit 6 set)
404	Power fail (DDE bit 3 set)
405	Request control
410	Query INTERRUPTED (QYE bit 2 set)
420	Query UNTERMINATED (QYE bit 2 set)
430	Query DEADLOCKED (QYE bit 2 set)
440	Query UNTERMINATED after indefinite response (QYE bit 2 set)

**Execution Warning**

The following table lists warning messages that do not interrupt the flow of command execution. These notify you that you may get unexpected results.

**Table 3-8: Execution Warning Messages (EXE Bit 4)**

<b>Code</b>	<b>Message</b>
500	Execution warning
510	String data too long, truncated
525	Parameter underrange
526	Parameter overrange
527	Parameter rounded
528	Parameter out of range
530	Data stop > start. Values swapped internally
531	Data stop > record length, Curve truncated
532	Curve data too long, Curve truncated
540	Measurement warning
541	Measurement warning, Low signal amplitude
542	Measurement warning, Unstable histogram
543	Measurement warning, Low resolution
544	Measurement warning, Uncertain edge
545	Measurement warning, Invalid minmax
546	Measurement warning, Need 3 edges
547	Measurement warning, Clipping positive/negative
548	Measurement warning, Clipping positive
549	Measurement warning, Clipping negative



**Internal Warning**

The following table shows internal errors that indicate an internal fault in the instrument.

**Table 3-9: Internal Warning Messages**

<b>Code</b>	<b>Message</b>
600	Internal warning



---

# Programming Examples

The DSA8300 comes with sample programs that illustrate methods that you can use to control the instrument from the GPIB interface.

The programs were written in Microsoft Visual C++, version 6.0, and were generated using the AppWizard program. Both program listings and source codes are included on the product software CD-ROM that comes with your instrument. A description of each program follows:

**Table 4-1: Example Programs**

<b>Name</b>	<b>Function</b>	<b>Module</b>
Meas	Performs a timing or an amplitude measurement on the selected waveform and displays the results.	Any 80E00
Wfm Transfer	Transfers the horizontal timebase and its vertical scale of the selected waveform to a file on your PC (controller).	Any 80E00
Capture	Captures the instrument screens and prints <sup>1</sup> it or saves it as a file on your instrument (not your controller).	Any 80E00
TDR	Displays the attached cable length in centimeters.	80E04 only
Mask	Makes an eye pattern evaluation against a mask, looking for a "stops after" condition you specify. Upon fulfillment of the stop after condition, the program can print <sup>1</sup> the result or can save it as a file on your instrument (not your controller).	Any 80C00 <sup>2</sup>

<sup>1</sup> To print, make sure that you have a default printer selected for your instrument before running this program.

<sup>2</sup> 80C01-CR module is preferred for use with this program.

- To Run**
1. Find the product-software CD-ROM included with your instrument, and insert it in the CD-ROM drive of your computer/controller.
  2. Navigate to the folder Samples > NI488.2 Samples on the CD. Find the self-extracting archive file named Samples.exe in that folder.
  3. Click the self-extracting archive to execute it. In the dialog that displays, accept the default installation directory or browse to your alternative location on your computer/controller. To run the programs, your computer/controller must be a PC-compatible system equipped with a National Instruments GPIB board and associated drivers.
  4. Ensure that:
    - the instrument is set as DEV1 with address of 1, and that the PC (controller) is set as GPIB0.
    - a valid signal is connected to the instrument.
  5. Navigate to the Sources directory contained in the Samples directory that you extracted.
  6. Click to execute the program that you wish to run. The program will display a dialog box from which you can set up and run the program.

---

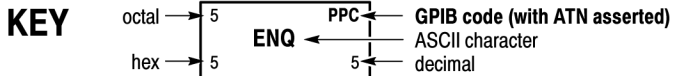
**NOTE.** *To reassign the name of your instrument, you can, enter your current device name into the appropriate edit box within each program (except the Mask program, where the board address is hard-coded to zero). You can also use IBCONF.EXE to reassign the name.*

---

- To Copy**
1. Do steps 1 through 3 of the procedure To Run the Sample Programs if you have not yet installed the Samples directory containing the programs.
  2. Navigate to the Programs directory contained in the Samples directory.
  3. Open the program file in your favorite editor or IDE, and copy the code you want into your programs.

# Appendix A: Character Set

B7 B6 B5 BITS B4 B3 B2 B1	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	20 DLE	40 SP	60 0	100 @	120 P	140 ' SA0	160 p SA16	0	10	20	30	40	50	60	70
0 0 0 1	1 SOH	21 DC1	41 !	61 1	101 A	121 Q	141 a SA1	161 q SA17	1	11	21	31	41	51	61	71
0 0 1 0	2 STX	22 DC2	42 "	62 2	102 B	122 R	142 b SA2	162 r SA18	2	12	22	32	42	52	62	72
0 0 1 1	3 ETX	23 DC3	43 #	63 3	103 C	123 S	143 c SA3	163 s SA19	3	13	23	33	43	53	63	73
0 1 0 0	4 EOT	24 DC4	44 \$	64 4	104 D	124 T	144 d SA4	164 t SA20	4	14	24	34	44	54	64	74
0 1 0 1	5 ENQ	25 NAK	45 %	65 5	105 E	125 U	145 e SA5	165 u SA21	5	15	25	35	45	55	65	75
0 1 1 0	6 ACK	26 SYN	46 &	66 6	106 F	126 V	146 f SA6	166 v SA22	6	16	26	36	46	56	66	76
0 1 1 1	7 BEL	27 ETB	47 ' (	67 7	107 G	127 W	147 g SA7	167 w SA23	7	17	27	37	47	57	67	77
1 0 0 0	8 BS	30 CAN	50 (	70 8	110 H	130 X	150 h SA8	170 x SA24	8	18	28	38	48	58	68	78
1 0 0 1	9 HT	31 EM	51 )	71 9	111 I	131 Y	151 i SA9	171 y SA25	9	19	29	39	49	59	69	79
1 0 1 0	A LF	32 SUB	52 *	72 :	112 J	132 Z	152 j SA10	172 z SA26	10	20	30	40	50	60	70	80
1 0 1 1	B VT	33 ESC	53 +	73 ;	113 K	133 [	153 k SA11	173 { SA27	11	21	31	41	51	61	71	81
1 1 0 0	C FF	34 FS	54 ,	74 <	114 L	134 \	154 l SA12	174   SA28	12	22	32	42	52	62	72	82
1 1 0 1	D CR	35 GS	55 -	75 =	115 M	135 ]	155 m SA13	175 } SA29	13	23	33	43	53	63	73	83
1 1 1 0	E SO	36 RS	56 .	76 >	116 N	136 ^	156 n SA14	176 ~ SA30	14	24	34	44	54	64	74	84
1 1 1 1	F SI	37 US	57 /	77 ?	117 O	137 -	157 o SA15	177 RUBOUT (DEL)	15	25	35	45	55	65	75	85
	ADDRESSED COMMANDS	UNIVERSAL COMMANDS	LISTEN ADDRESSES				TALK ADDRESSES				SECONDARY ADDRESSES OR COMMANDS					



**Tektronix**  
 REF: ANSI STD X3.4-1977  
 IEEE STD 488.1-1987  
 ISO STD 646-2973



# Appendix B: Reserved Words

This is a list of reserved words for your instrument. Capital letters identify the required minimum spelling. For the most robust code, use the full spelling, since spelling rules may change over time and among instrument models.

*CLS	AVERage	COMPensate	EITHer
*ESE	AVGComp	CONDition	EMPH7
*ESR	Amp	CONTRol	EMPH8
*IDN	BACKground	COUNT	EMPHasis
*LRN	BACKward	CPRI7373	ENABle
*OPC	BANDwidth	CRC	ENCdg
*OPT	BELL	CREATEStandard	ENET100B_BX10
*PSC	BIN7	CRMS	ENET100B_LX10
*RST	BIN8	CROSSHair	ENET103R4
*SRE	BINary	CURRentcount	ENET103R10
*STB	BITRate	CURSOR1	ENET257R4
*WAI	BITS	CURSOR2	ENET1000B_KX
0	BITTime	CURSReadout	ENET10313
1	BIT_Nr	CURSOR	ENET10GB_LRM
80SJNB	BMP	CURVe	ENET11096
ABSolute	BN_Fmt	CUSTOM	ENET1250
ACQHWver	BOX	CUSTOMmask	ENET2500
ACQWfms	BOXPcnt	DARKLev	ENET3125
ACQuire	BURst	DATA	ENET9953
ACRMs	BUSY	DATARate	ENET41250
ACTion	BYT_Nr	DATE	ENV
ACTivate	BYT_Or	DATA	ENVElope
ALL	C1C2	DATE	ERCAL
ALLEV	C3C4	DB	EVENT
AMPLitude	C5C6	DCCALibrator	EVMsg
ANNOTations	C7C8	DCRATio	EVQty
AOPTPWRDBM	CALibrate	DEFine	EXECute
AOPTpwr	CAPacitance	DELETETandard	EXPort
AOPT	CAREa	DELEte	EXPrev
APPLication	CENTERed	DELTA	EXT10MHZref
AREa	CH	DELay	EXTAtten
ASCIi	CHAR	DESE	EXTDirect
ATARXG1	CHMAIn	DESTination	EXTINCTCAL
ATARXG2	CHx	DESkew	EXTINCTDB
ATARXG3	CLEar	DEVICEName	EXTINCTPCT
ATATXG1	CLKRAnge	DIElectric	EXTINCTRATIo
ATATXG2	CLKRECovery	DIFfpreset	EXTPrescaler
ATATXG3	CLKRec	DIRection	EXTender
AUTO	CLKUser	DISTDuty	EYE
AUTOPosition	CLOCKAmplitude	DISTance	EYECross
AUTOSConfigdev	CLOCKOutput	DISPlay	EYEHeight
AUTOSet	CMEan	DYNamic	EYEOfactor
AUTOSync	COLOR	EDGEDEnSITYMode	EYEWIdth
AUTOMatic	COLOR	EDGEDEnSity	EYEWindow
AUTO	COLOR	EDGE	FACTory

FACTory	FREQUency	JITter	MODE
FALL	FREerun	JPEG	MODElnum
FC1063	FT	KEY	MODE
FC1063E	FULl	LEVCRoss	MSB
FC11317	FUNCTion	LEVEl	NCross
FC133	GAIN	LINEAr	NDUty
FC133E	GATED	LINear	NOISe
FC2125	GATING	LIST	NOMEDgedensity
FC2125E_ABR	GRADMethoD	LISTValue	NOMFrequency
FC2125E_ABT	GRADing	LOCK	NOMinal
FC2125E_AGR	GRaticule	LOCKCount	NONDestruct
FC2125E_AGT	GRId	LOCKExt	NONEMAIN
FC266	GROUp	LOCKInt	NONE
FC266E	HARDCopy	LOCKMode	NORMal
FC3188	HBARS	LOCKRange	NOVershoot
FC4250	HDELTA	LOCKState	NR1
FC4250E_ABR	HDIVS	LOCK	NR3
FC4250E_ABT	HEADER	LODeLta	NR_Pt
FC4250E_AGR	HIDeLta	LOG	NUMAVg
FC4250E_AGT	HIFreq	LOOPBandwidth	NUMavg
FC531	HIGH	LOW	NUMerictxt
FC531E	HILOW	LSB	NWIdth
FC8500	HISTHits	MAG1	OBSAI6144
FC8500D	HISTReadout	MAG2	OC1
FC8500E_ABR	HISTWaveform	MAGnify	OC3
FC8500E_ABT	HISTwfms	MAIn	OC9
FC8500E_AGR	HISTogram	MAInframe	OC12
FC8500E_AGT	HITS	MANuaL	OC18
FC8500FINAL	HOLDoff	MARgin	OC24
FC10519	HORizontal	MASK	OC36
FC14025	HOSTinfo	MASKHits	OC48
FC8GFCDREV6	HPOS	MASKReadout	OC96
FEC2666	HWRev	MASKSample	OC192
FEC10664	ID	MASKSamples	OC768
FEC10709	IMPedance	MASKTHits	OFF
FEC12500	IMPort	MASKTOTAlhit	OFFSet
FEC42657	IN	MASKWaveform	Ohm
FEC43018	INCident	MASKwfms	OMA
FILENAME	INF2500	MATH	ON
FILESystem	INF5000	MATHX	ONTime
FILTer	INF14063	MAXimum	ON_Lock
FOREground	INF25781	MEAN	OPERating
FORMat	INFIE25	MEAS	OPTGAIN
FORward	INFIniband	MEASBar	OTU27952
FP	INFPersist	MEASReadout	PCIEXPRESS50_Rcv
FPAnel	INIT	MEASurement	PCIEXPRESS_Rcv
FPBinary	INKSaver	MEDIAN	PCIX2500
FPELCOUT	INPUT_DATA	METHod	PCIX6250
FPGarev	INTClk	MID	PCROSS
FRAMEcycle	INTERPolat	MIDref	PCTCROSS
FRAMEScan	INTRate	MINMax	PCTPos
FRame	INTensity	MINUS	PDUty
FREQ	INVert	MINimum	PEAKHits



PEAKing	RESuLts	SIZE	TYPE
PERCent	RI	SLOPe	UNDEFIned
PERIod	RIBinary	SLOPe	UNDO
PERSistence	RIO_SERIAL1G	SNAP	UNIQueid
PHASEERRLimit	RIO_SERIAL2G	SNRatio	UNItS
PHASEERRP2P	RIO_SERIAL3G	SOURce	UNLock
PHASEERRRMS	RISE	SOURce	UPDATEInfo
PHASE	RISe	SPReadsheetcsv	USERMask
PHASeref	RISetime	SRIBinary	USER
PK2Pk	RMS	SSCTrigger	VALue
PKPKJitter	RMSJitter	STANDARDList	VARPersist
PKPKNoise	RMSNoise	STANDARD	VBARS
PKTOPK	RUN	STARTBit	VDELTA
PLENRAnge	RUNStop	START	VDIVS
PLENgtH	RZ	STATE	VERBoSe
PLUS	Rho	STATISTICS	VERTical
PNG	S	STATISTICS	VIEW
POINTSPcnt	SAMPles	STATUS	VSR5_3318
POINts	SAMplE	STDdev	VoIt
POLArity	SAS3_0_SATA	STEP	WAVFrm
POS	SAS3_0_XR	STOP	WAVEform
POSITION	SAS3_0_XR_AASJ	STOPAfter	WAVEforms
POVershoot	SATA1500	STYle	WEIghting
POWER_ON	SATA1500m	SUBCLOCKAmpl	WFID
POWERups	SATA3000	SUBCLOCKDIVList	WFM
PRAnge	SATA6000	SUBCLOCKDiv	WFMDB
PRESET	SAVEFile	SUBCLOCKOutput	WFMInpre
PRINTScreen	SAVEScreen	SUPRSDB	WFMLabel
PROCHWver	SAVEWaveform	SUPRSPCT	WFMOutpre
PROPerties	SAVEWfm	SUPRSRATIO	WFMReadout
PRObe	SAVe	SWRev	WLEngth
PSYNC	SCANBits	SWVersion	Watt
PT_Fmt	SCALE	SYNC	XAUI3125
PULSE	SCONfigdevice	SYSTEM	XAUI6250
PULSESym	SElect	TARGA	XAUIFar
PVELocity	SERIALnum	TBMode	XAUINear
PWIDth	SERIALnum	TDR	XAUIRFar
QFACTOR	SET	TEKPDriver	XAUIRNear
RANGE	SETDefault	TEKPVersion	XFI9950_RAD
RBIT	SETLevel	TEMPerature	XFI9950_RHC
RCONfigdevice	SETUP_1	TIFF	XFI9950_RMCP
READFile	SETUP_2	TIMCROSS	XFI9950_TAA
RECAll	SETUP_3	TIMEOUT	XFI9950_THB
RECOrdlength	SETUP_4	TIME	XFI9950_TMBP
REF	SETUp	TIMEbase	XINcr
REFMAIn	SFEC12500	TOFPoint	XMUIt
REFPoint	SFPBinary	TOLPoint	XOFF
REFXMAIn	SHIFted	TOTAL	XUNit
RELative	SHORTTerm	TOTALontime	XZerO
RELock	SHOWVector	TOUCH	Y
RESET	SIGMA	TRACK	YMUIt
RESETLockcount	SIGType	TRIGProbe	YOFF
RESOlution	SINX	TRIGger	YSCALE

YUNit

YZero

# Appendix C: Factory Default Setup Values

The following tables list the default setup values by command group. These are the values that the instrument sets when you press the DEFAULT SETUP front-panel button or send the FACTory command. Only those commands that have values set by the DEFAULT SETUP function are listed in these tables.

## Acquisition Default Values

The following table lists the Acquisition factory default setup values.

Command	Default Values
ACQuire:MODE	SAMPLE
ACQuire:NUMAVg	16
ACQuire:SAVEFile:SAVEScreen	“ “
ACQuire:SAVEFile:SAVEWfm	“ “
ACQuire:STATE	1 (ON)
ACQuire:STOPAfter:ACTion	NONE
ACQuire:STOPAfter:BELL	0 (OFF)
ACQuire:STOPAfter:CONDition	ACQWFMS
ACQuire:STOPAfter:COUNT	1
ACQuire:STOPAfter:MODE	RUNSTOP

## Cursor Default Values

The following table lists the Cursor factory default setup values.

Command	Default Values
CURSor:CURSor<x>:COLOR	255
CURSor:CURSor<x>:SOURce	CH1,MAIN
CURSor:FUNCTion	0 (OFF)
CURSor:HBArs:POSition<x>	Cursor 1 = -1.621616E-001 Cursor 2 = 1.702702E-001
CURSor:SElect	NONE
CURSor:VBArS:POSition<x>	Cursor 1 = 1.925E-008 Cursor 2 = 2.065E-008
CURSor:WAVEform:POSition<x>	Cursor 1 = 1.9265E-008 Cursor 2 = 2.065E-008

## Display Default Values

The following table lists the Display factory default setup values.

Command	Default Values
DISplay:COLor:BACKground	0 (black)
DISplay:COLor:CURSor<x>	Cursor 1 = 255 (red) Cursor 2 = 255 (red)

Command	Default Values
DISplay:COLor:FOREground	16777215 (white)
DISplay:COLor:HIStoqram	16776960 (aqua)
DISplay:COLor:MASK	16711935 (fuchsia)
DISplay:CURSRReadout	0 (OFF)
DISplay:GRAticule:STYLE	FULL
DISplay:HISTRReadout	0 (OFF)
DISplay:INTERPolat	SINX
DISplay:MASKReadout	0 (OFF)
DISplay:MEASBar	0 (OFF)
DISplay:MEASReadout	0 (OFF)
DISplay:PERSiistence	2.000E-001
DISplay:SHOWVector	0 (OFF)
DISplay:STYle	NORMAL
DISplay:WFMReadout	0 (OFF)

### Histogram Default Values

The following table lists the Histogram factory default setup values.

Command	Default Values
HIStoqram:BOX	0 (OFF)
HIStoqram:BOXPcnt	3.000E+001 2.510E+001 7.000E+001 7.520E+001
HIStoqram:COLOr	16776960 (aqua)
HIStoqram:DISPlay	0 (OFF)
HIStoqram:ENABle	0 (OFF)
HIStoqram:MODe	HORIZONTAL
HIStoqram:SIZe	2
HIStoqram:SOUrce	CH1,MAIN
HIStoqram:TYPE	LINEAR
HIStoqram:WFMDB:STATE	0 (OFF)

### Horizontal Default Values

The following table lists the Horizontal factory default setup values.

Command	Default Values
HORizontal:BITS:BITRate	2.488E+009
HORizontal:BITS:STANdard	NONE
HORizontal:DISPlayscale:BITS	PERScreen

Command	Default Values
HORizontal:DISPlayscale:DISTance	PERDivision
HORizontal:DISPlayscale:SEConds	PERDivision
HORizontal:DISTance:DIElectric	1
HORizontal:DISTance:PVELocity	2.998E+10
HORizontal:FRAMescan:SCANBits	127
HORizontal:FRAMescan:STATE	0 (OFF)
HORizontal:MAGnify<x>:POSition	1.975E-008
HORizontal:MAGnify<x>:RECOrdlength	4000
HORizontal:MAGnify<x>:REFPoint	5.000E-001
HORizontal:MAGnify<x>:SCALE	5.000E-11
HORizontal:MAGnify<x>:VIEW	0 (OFF)
HORizontal:MAIn:POSition	2.150E-008
HORizontal:MAIn:RECOrdlength	4000
HORizontal:MAIn:REFPoint	5.000E-001
HORizontal:MAIn:SCALE	5.000E-10
HORizontal:UNIts	S

### Mask Default Values

The following table lists the Mask factory default setup values.

Command	Default Values
MASK:AUTOSet:MODE	MANUAL
MASK:COLOr	16711935 (fuchsia)
MASK:COUNT:STATE	0 (OFF)
MASK:DISPlay	0 (OFF)
MASK:MARGin:PERCent	5.000E+000
MASK:MARGin:STATE	0 (OFF)
MASK:SOUrce	CH1,MAIN
MASK:STANdard	NONE
MASK:WFMDB:STATE	0 (OFF)

### Math Default Values

The following table lists the Math factory default setup values.

Command	Default Values
MATH<x>:DEFine	" "
MATH<x>:FILTer:MODE	CENTERED
MATH<x>:FILTer:RISetime	1.000E+009
MATH<x>:NUMavg	2

Command	Default Values
MATH<x>:POSition	0.00E+000
MATH<x>:SCAle	1.000E+000

**Measurement Default Values**

The following table lists the Measurement factory default setup values.

Command	Default Values
MEASUrement:ANNOtations:STATE	0 (OFF)
MEASUrement:MEAS<x>:EYEWindow	20
MEASUrement:MEAS<x>:GATing:STATE	0 (OFF)
MEASUrement:MEAS<x>:JITter	EYECROSS
MEASUrement:MEAS<x>:NOISe	HIGH
MEASUrement:MEAS<x>:REFLevel<x>:ABSolute:HIGH	REFLEVEL 1 = 0.000E+000
MEASUrement:MEAS<x>:REFLevel<x>:ABSolute:LOW	REFLEVEL 1 = 0.000E+000
MEASUrement:MEAS<x>:REFLevel<x>:ABSolute:MID	REFLEVEL 1 = 0.000E+000
MEASUrement:MEAS<x>:REFLevel<x>:RELative:HIGH	REFLEVEL 1 = 9.000E+001
MEASUrement:MEAS<x>:REFLevel<x>:RELative:LOW	REFLEVEL 1 = 1.000E+001
MEASUrement:MEAS<x>:REFLevel<x>:RELative:MID	REFLEVEL 1 = 5.000E+001
MEASUrement:MEAS<x>:REFLevel<x>:METHod	RELATIVE
MEASUrement:MEAS<x>:SOUrce<x>:EDGE:DIRection	FORWARD
MEASUrement:MEAS<x>:SOUrce<x>:EDGE:SLOPe	RISE
MEASUrement:MEAS<x>:SOUrce<x>:GATE<x>:POS	Gate 1 = 1.900E-008 Gate 2 = 2.399E-008
MEASUrement:MEAS<x>:SOUrce<x>:HILow:METHod	AUTO
MEASUrement:MEAS<x>:SOUrce<x>:HILow:TRACK: HIGH:ENABle	0 (OFF)
MEASUrement:MEAS<x>:SOUrce<x>:HILow:TRACK: HIGH:VALue	0.000e+000
MEASUrement:MEAS<x>:SOUrce<x>:HILow:TRACK: LOW:ENABle	0 (OFF)
MEASUrement:MEAS<x>:SOUrce<x>:HILow:TRACK: LOW:VALue	0.000E+000
MEASUrement:MEAS<x>:SOUrce<x>:WFM	CH1,MAIN
MEASUrement:MEAS<x>:SOUrce<x>:WFMDB:SIGType	EYE
MEASUrement:MEAS<x>:SOUrce<x>:WFMDB:STATE	0 (OFF)
MEASUrement:MEAS<x>:STATE	0 (OFF)
MEASUrement:MEAS<x>:TYPe	UNDEFINED
MEASUrement:STATIstics:ENABle	0 (OFF)
MEASUrement:STATIstics:WEIghting	32

**Miscellaneous Default Values**

The following table lists the Miscellaneous factory default setup values.

Command	Default Values
AUTOSet:TYPE	EYE
HARDCopy:FORMat	BMP
HARDCopy:INKSaver	1(ON)
HEADer	0 (OFF)
LOCK	FPANEL
VERBose	1 (ON)

**Status and Error Values**

The following table lists the Status and Error factory default setup values.

Command	Default Values
DESE	255
*ESE	0
*SRE	0

**TDR Default Values**

The following table lists the TDR factory default setup values.

Command	Default Values
TDR:CH<x>:STEP:DESkeW	0.000E+000
TDR:CH<x>:STEP:POLarity	PLUS
TDR:CH<x>:STEP:STATE	0 (OFF)
TDR:CH<x>:UNIts	RHO
TDR:INTRate	200

**Trigger Default Values**

The following table lists the Trigger factory default setup values.

Command	Default Values
TRIGger:CH<x>:CLKRec:VALue	NONE
TRIGger:CLKRec:SOURce	NONE
TRIGger:HIFreq	0 (OFF)
TRIGger:HOLDoff	5.000E-006
TRIGger:LEVel	0.000E+000
TRIGger:MODE	EYE
TRIGger:SLOpe	RISE
TRIGger:SOURce	EXTPRESCALER

**Vertical Default Values**

The following table lists the Vertical factory default setup values.

Command	Default Values
CH<x>:BANdwidth:VALue	1.000E+032 (forces selection of a module's highest available bandwidth)
CH<x>:DESkew	0.000E+000
CH<x>:EXTAtten:MODE	LINEAR
CH<x>:EXTAtten:VALue	1.00E+000
CH<x>:FILTer:VALue	NONE
CH<x>:OFFSet	0.000E+000
CH<x>:POSition	0.000E+000
CH<x>:SCAle	1.000E-000
CH<x>:UNIts	AUTO
CURVe	ASCII
DATa:DESTination	REF1
DATa:ENCdg	ASCII
DATa:SOURce	CH1
DATa:STARt	1
DATa:STOP	1000
REF<x>:POSition	0.00
REF<x>:SCAle	1.000
SElect:CH<x>	0
SElect:CONTROl	NONE,MAIN
SElect:MATH<x>	0
SElect:REF<x>	0

### Waveform Database Values

The following table lists the Waveform Database factory default setup values.

Command	Default Values
WFMDB:DISPlay:EMPHasis	50
WFMDB:DISPlay:GRADing	COLOR
WFMDB:DISPlay:GRADMethod	BIN8
WFMDB:DISPlay:INVert	0 (OFF)
WFMDB:WFMDB<x>:DISplay	0 (OFF)
WFMDB:WFMDB<x>:ENABle	0 (OFF)
WFMDB:WFMDB<x>:SOURce	CH1,MAIN

### Waveform Transfer Values

The following table lists the Waveform Transfer factory default setup values.



Command	Default Values
WFMinpre:BIT_Nr?	32
WFMinpre:BN_Fmt	RI
WFMinpre:BYT_Nr?	4
WFMinpre:BYT_Or	LSB
WFMinpre:ENCdg	ASCII
WFMinpre:NR_Pt	1000
WFMinpre:PT_Fmt	Y
WFMinpre:WFMLabel	NA
WFMinpre:XINcr	0.000E+000
WFMinpre:XMUlt	1.000E-010
WFMinpre:XOFF	21.50000E-9
WFMinpre:XUNit	" "
WFMinpre:XZEro	1.600E-0008
WFMinpre:YMUlt	1.000E-001
WFMinpre:YOFF	0
WFMinpre:YSCALE	0.00E+000
WFMinpre:YUNit	" "
WFMinpre:YZEro	0.00E+000
WFMOupre:BIT_Nr?	32
WFMOupre:BN_Fmt	RI
WFMOupre:BYT_Nr?	4
WFMOupre:BYT_Or	LSB
WFMOupre:ENCdg	ASCII
WFMOupre:WFMLabel?	" "



# Appendix D: GPIB Interface Specifications

This topic describes details of the GPIB remote interface of the instrument. Normally, you will not need this information to use the instrument, but this information may be useful if you are connecting to controllers with unusual configurations.

## Interface Messages

The following table shows the standard interface messages that the instrument supports.

**Table D-1: Standard Interface Messages**

Message	Interface
DCL	Yes
GET	Yes
GTL	Yes
LLO	Yes
PPC	No
PPD	No
PPE	No
PPU	No
SDC	Yes
SPD	Yes
SPE	Yes
TCT	No
UNL	Yes
UNT	Yes
Listen Addresses	Yes
Talk Addresses	Yes

## GPIB Functions

The following table lists the interface functions and electrical functions subsets supported by this instrument with brief descriptions of each.

**Table D-2: Interface Functions**

Interface Function	Subset	Meaning
Acceptor Handshake	AH1	The instrument can receive multiline messages across the interface from other devices.
Controller	C1	Controller capability, the instrument can control other devices.

**Table D-2: Interface Functions (cont.)**

<b>Interface Function</b>	<b>Subset</b>	<b>Meaning</b>
Device Clear	DC1	The instrument can respond to both the DCL (Device Clear) interface message and to the Selected Device Clear (SDC) interface message when the instrument is listen-addressed.
Device Trigger	DT1	Device Trigger capability, the instrument does respond to the GET (Group Execute Trigger) interface message.
Electrical	E2	The instrument uses tri-state buffers, which are optimal for high-speed data transfer.
Listener	L4	The instrument becomes a listener when it detects the listen address being sent over the bus with the ATN line asserted. The instrument ceases to be a listener and becomes a talker when it detects the talk address being sent over the bus with the ATN line asserted.
Parallel Poll	PP0	No Parallel Poll Capability, the instrument does not respond to PPC (Parallel Poll Configure), PPO (Parallel POI Disable), PPE (Parallel Poll Enable), or PPU (Parallel Poll Unconfigure) interface messages, nor does it send a status message when the ATN and EOI lines are asserted simultaneously.
Remote/Local	RL1	The instrument can respond to both GTL (Go To Local) and LLO (Local Lock Out) interface messages.
Service Request	SR1	The instrument can assert the SRQ line to notify the controller in charge that it requires service.
Source Handshake	SH1	The instrument can initiate multilines messages to send across the interface to other devices.
Talker	T5	The instrument becomes a talker when it detects the talk address being sent over the bus with the ATN line asserted. The instrument ceases to be a talker and becomes a listener when it detects the listen address being sent over the bus with the ATN line asserted. The instrument also ceases to be a talker when it detects the talk address of another device being sent over the data lines with ATN asserted.

# Index

## Symbols and Numbers

, 2-334

### A

ACQUIRE:CURRENTCOUNT:ACQWFMS?, 2-49  
ACQUIRE:CURRENTCOUNT:HISTHITS?, 2-50  
ACQUIRE:CURRENTCOUNT:HISTWFMS?, 2-50  
ACQUIRE:CURRENTCOUNT:MASKHITS<x>?, 2-51  
ACQUIRE:CURRENTCOUNT:MASKSAMPLES?, 2-51  
ACQUIRE:CURRENTCOUNT:MASKTHITS?, 2-52  
ACQUIRE:CURRENTCOUNT:MASKWFMS?, 2-52  
ACQUIRE:DATA:CLEAR, 2-53  
ACQUIRE:MODE, 2-54  
ACQUIRE:NUMAVG, 2-55  
ACQUIRE:SAVEFILE:SAVESCREEN, 2-56  
ACQUIRE:SAVEFILE:SAVEWFM, 2-56  
ACQUIRE:STATE, 2-57  
ACQUIRE:STOPAFTER:ACTION, 2-58  
ACQUIRE:STOPAFTER:BELL, 2-59  
ACQUIRE:STOPAFTER:CONDITION, 2-60  
ACQUIRE:STOPAFTER:COUNT, 2-62  
ACQUIRE:STOPAFTER:MODE, 2-62  
ACQUIRE:STOPAFTER?, 2-58  
ACQUIRE?, 2-49  
ALLEV?, 2-63  
APPLICATION:ACTIVATE, 2-64  
AUTOSet, 2-65  
AUTOSet:HORIZONTAL, 2-66  
AUTOSet:STOP, 2-66  
AUTOSet:TDR:INCIDENT, 2-67  
AUTOSet:TRIGGER, 2-67  
AUTOSet:TYPE, 2-68  
AUTOSet:UNDO, 2-69  
AUTOSet:VERTICAL, 2-70

### B

BUSY?, 2-70

### C

CALIBRATE:DATE:CH<x>?, 2-71  
CALIBRATE:DATE:MAINFRAFRAME?, 2-72  
CALIBRATE:DCCALIBRATOR, 2-72

CALIBRATE:HOSTINFO:CH<x>?, 2-72  
CALIBRATE:LOCK:STATUS?, 2-73  
CALIBRATE:STATUS:CH<x>?, 2-73  
CALIBRATE:STATUS:MAINFRAFRAME?, 2-73  
CALIBRATE:TEMPERATURE:CH<x>?, 2-74  
CALIBRATE:TEMPERATURE:MAINFRAFRAME?, 2-74  
CALIBRATE:UPDATEINFO:ALLMODULES, 2-75  
CALIBRATE:UPDATEINFO:CH<x>, 2-75  
CALIBRATE:UPDATEINFO:MAINFRAFRAME, 2-75  
CH<x>:BANDWIDTH:LIST?, 2-76  
CH<x>:BANDWIDTH:VALUE, 2-77  
CH<x>:DELAY, 2-77  
CH<x>:DESKW, 2-78  
CH<x>:ERCAL?, 2-79  
CH<x>:EXTATTEN:MODE, 2-79  
CH<x>:EXTATTEN:VALUE, 2-80  
CH<x>:FILTER:LIST?, 2-81  
CH<x>:FILTER:VALUE, 2-81  
CH<x>:OFFSET, 2-84  
CH<x>:POSITION, 2-85  
CH<x>:SCALE, 2-86  
CH<x>:UNITS, 2-87  
CH<x>:WFMLABEL, 2-88  
CH<x>:WLENGTH:LIST?, 2-88  
CH<x>:WLENGTH:VALUE, 2-89  
CH<x>?, 2-76  
\*CLS, 2-90  
Command Groups, 2-11  
COMPENSATE:ALLMODULES, 2-90  
COMPENSATE:CH<x>, 2-91  
COMPENSATE:DARKLEV:CH<x>, 2-91  
COMPENSATE:DATE:CH<x>?, 2-92  
COMPENSATE:DATE:MAINFRAFRAME?, 2-92  
COMPENSATE:MAINFRAFRAME, 2-93  
COMPENSATE:OPTGAIN:CH<x>, 2-93  
COMPENSATE:RECALL:FACTORY:ALLMODULES, 2-94  
COMPENSATE:RECALL:FACTORY:CH<x>, 2-94  
COMPENSATE:RECALL:FACTORY:MAINFRAFRAME, 2-94  
COMPENSATE:RECALL:USER:ALLMODULES, 2-95  
COMPENSATE:RECALL:USER:CH<x>, 2-95  
COMPENSATE:RECALL:USER:MAINFRAFRAME, 2-95  
COMPENSATE:RESULTS:VERBOSE?, 2-96  
COMPENSATE:RESULTS?, 2-96  
COMPENSATE:SAVE:USER:ALLMODULES, 2-96

COMPensate:SAVe:USER:CH<x>, 2-97  
COMPensate:SAVe:USER:MAInframe, 2-97  
COMPensate:STATus:CH<x>?, 2-97  
COMPensate:STATus:MAInframe?, 2-98  
COMPensate:TEMPerature:CH<x>?, 2-98  
COMPensate:TEMPerature:MAInframe?, 2-98  
CURSor:CURSor<x>:COLOR, 2-99  
CURSor:CURSor<x>:SOUrce, 2-100  
CURSor:FUNCTion, 2-101  
CURSor:HBArS:DELTA?, 2-102  
CURSor:HBArS:POSition<x>, 2-103  
CURSor:HBArS?, 2-102  
CURSor:SELEct, 2-103  
CURSor:VBArS:DELTA?, 2-104  
CURSor:VBArS:POSition<x>, 2-105  
CURSor:VBArS?, 2-104  
CURSor:WAVEform:HDELTA?, 2-106  
CURSor:WAVEform:HPOS<x>?, 2-106  
CURSor:WAVEform:POSition<x>, 2-106  
CURSor:WAVEform:VDELTA?, 2-107  
CURSor:WAVEform?, 2-105  
CURSor?, 2-99  
CURVe, 2-107

## D

DATA, 2-109  
DATA:DESTination, 2-109  
DATA:ENCdg, 2-110  
DATA:SOUrce, 2-111  
DATA:STARt, 2-111  
DATA:STOP, 2-112  
DATA:TIMEbase, 2-113  
DATE, 2-114  
DELEte:WAVEform, 2-114  
DESE, 2-115  
DISplay:COLor:BACKground, 2-117  
DISplay:COLor:CURSor<x>, 2-118  
DISplay:COLor:FOREground, 2-119  
DISplay:COLor:HISTogram, 2-121  
DISplay:COLor:MASK, 2-122  
DISplay:COLor?, 2-116  
DISplay:CURSReadout, 2-123  
DISplay:DATE, 2-123  
DISplay:GRAticule:HDIVS?, 2-124  
DISplay:GRAticule:STYLE, 2-125  
DISplay:GRAticule:VDIVS?, 2-125  
DISplay:GRAticule?, 2-124

DISplay:HISTReadout, 2-125  
DISplay:INTERPolat, 2-126  
DISplay:MASKReadout, 2-127  
DISplay:MEASBar, 2-128  
DISplay:MEASReadout, 2-128  
DISplay:PERsistence, 2-129  
DISplay:SHOWVector, 2-129  
DISplay:STYle, 2-130  
DISplay:WFMReadout, 2-131  
DISplay?, 2-116

## E

\*ESE, 2-131  
\*ESR?, 2-132  
EVENT?, 2-132  
EVMsg?, 2-133  
EVQty?, 2-133  
EXPort, 2-134  
EXPort:TYPE, 2-135

## F

FACTory, 2-135  
FILESystem:READFile?, 2-136

## H

HARDCopy, 2-136  
HARDCopy:FILEName, 2-137  
HARDCopy:FORMat, 2-138  
HARDCopy:INKSaver, 2-139  
HEADer, 2-140  
HISTogram:BOX, 2-141  
HISTogram:BOXPent, 2-142  
HISTogram:COLOr, 2-142  
HISTogram:COUNt, 2-144  
HISTogram:DISplay, 2-144  
HISTogram:ENABle, 2-145  
HISTogram:MODE, 2-145  
HISTogram:SIZE, 2-146  
HISTogram:SOUrce, 2-146  
HISTogram:STATistics:HITS?, 2-148  
HISTogram:STATistics:MEAN?, 2-148  
HISTogram:STATistics:MEDIAN?, 2-149  
HISTogram:STATistics:PEAKHits?, 2-149  
HISTogram:STATistics:PKTOPK?, 2-149  
HISTogram:STATistics:SIGMA<x>?, 2-150  
HISTogram:STATistics:STDdev?, 2-150

- HIsTogram:STATistics:WAVEforms?, 2-150  
 HIsTogram:STATistics?, 2-147  
 HIsTogram:TYPE, 2-151  
 HIsTogram:WFMDB:STATE, 2-151  
 HIsTogram?, 2-140  
 HORizontal:BITS:BITRate, 2-153  
 HORizontal:BITS:STANdard, 2-153  
 HORizontal:DISPlayscale:BITS, 2-158  
 HORizontal:DISPlayscale:DISTance, 2-159  
 HORizontal:DISPlayscale:SEConds, 2-160  
 HORizontal:DISTance:DIElectric, 2-160  
 HORizontal:DISTance:PVELocity, 2-161  
 HORizontal:EXT10MHZref:FREQ, 2-162  
 HORizontal:FRAMescan:RESET, 2-163  
 HORizontal:FRAMescan:SCANBits, 2-163  
 HORizontal:FRAMescan:STATE, 2-163  
 HORizontal:MAGnify<x>:POSition, 2-165  
 HORizontal:MAGnify<x>:RECOrdlength, 2-166  
 HORizontal:MAGnify<x>:REFPoint, 2-167  
 HORizontal:MAGnify<x>:RESolution?, 2-168  
 HORizontal:MAGnify<x>:SCALE, 2-169  
 HORizontal:MAGnify<x>:TOFPoint?, 2-169  
 HORizontal:MAGnify<x>:TOLPoint?, 2-170  
 HORizontal:MAGnify<x>:VIEW, 2-171  
 HORizontal:MAGnify<x>?, 2-164  
 HORizontal:MAIn:BITS:POSition, 2-172  
 HORizontal:MAIn:BITS:SCALE, 2-173  
 HORizontal:MAIn:POSition, 2-173  
 HORizontal:MAIn:RECOrdlength, 2-175  
 HORizontal:MAIn:REFPoint, 2-176  
 HORizontal:MAIn:RESolution?, 2-177  
 HORizontal:MAIn:SCALE, 2-177  
 HORizontal:MAIn:TOFPoint?, 2-178  
 HORizontal:MAIn:TOLPoint?, 2-179  
 HORizontal:MAIn?, 2-172  
 HORizontal:MATH<x>:MAGnify<x>:  
     POSition?, 2-179  
 HORizontal:MATH<x>:MAGnify<x>:  
     RECOrdlength?, 2-180  
 HORizontal:MATH<x>:MAGnify<x>:  
     REFPoint?, 2-180  
 HORizontal:MATH<x>:MAGnify<x>:  
     RESolution?, 2-181  
 HORizontal:MATH<x>:MAGnify<x>:SCALE?, 2-181  
 HORizontal:MATH<x>:MAGnify<x>:  
     TOFPoint?, 2-182  
 HORizontal:MATH<x>:MAGnify<x>:  
     TOLPoint?, 2-182  
 HORizontal:MATH<x>:MAIn:POSition?, 2-183  
 HORizontal:MATH<x>:MAIn:RECOrdlength?, 2-183  
 HORizontal:MATH<x>:MAIn:REFPoint?, 2-184  
 HORizontal:MATH<x>:MAIn:RESolution?, 2-185  
 HORizontal:MATH<x>:MAIn:SCALE?, 2-185  
 HORizontal:MATH<x>:MAIn:TOFPoint?, 2-185  
 HORizontal:MATH<x>:MAIn:TOLPoint?, 2-186  
 HORizontal:REF<x>:MAGnify<x>:POSition?, 2-186  
 HORizontal:REF<x>:MAGnify<x>:  
     RECOrdlength?, 2-187  
 HORizontal:REF<x>:MAGnify<x>:  
     REFPoint?, 2-187  
 HORizontal:REF<x>:MAGnify<x>:  
     RESolution?, 2-188  
 HORizontal:REF<x>:MAGnify<x>:SCALE?, 2-188  
 HORizontal:REF<x>:MAGnify<x>:  
     TOFPoint?, 2-189  
 HORizontal:REF<x>:MAGnify<x>:  
     TOLPoint?, 2-189  
 HORizontal:REF<x>:MAIn:POSition?, 2-190  
 HORizontal:REF<x>:MAIn:RECOrdlength?, 2-190  
 HORizontal:REF<x>:MAIn:REFPoint?, 2-191  
 HORizontal:REF<x>:MAIn:RESolution?, 2-191  
 HORizontal:REF<x>:MAIn:SCALE?, 2-192  
 HORizontal:REF<x>:MAIn:TOFPoint?, 2-192  
 HORizontal:REF<x>:MAIn:TOLPoint?, 2-193  
 HORizontal:UNIts, 2-193  
 HORizontal?, 2-152
- I**  
 ID?, 2-194  
 \*IDN?, 2-194  
 IMPort:CUSTommask, 2-195
- L**  
 LOCK, 2-195  
 \*LRN?, 2-196
- M**  
 MASK:AUTOSet:HILow:METhod, 2-198  
 MASK:AUTOSet:MODE, 2-199  
 MASK:COLOR, 2-199  
 MASK:COUNt, 2-200  
 MASK:COUNt:SAMPles?, 2-201

MASK:COUNT:STATE, 2-201  
 MASK:COUNT:TOTal?, 2-202  
 MASK:COUNT:WAVEforms?, 2-202  
 MASK:DISplay, 2-203  
 MASK:MARgin:PERCent, 2-203  
 MASK:MARgin:STATE, 2-204  
 MASK:MASK<x>, 2-205  
 MASK:MASK<x>:COUNT?, 2-205  
 MASK:MASK<x>:NR\_Pt?, 2-206  
 MASK:MASK<x>:POINts, 2-206  
 MASK:MASK<x>:POINTSPcnt, 2-207  
 MASK:SOURce, 2-208  
 MASK:STANdard, 2-209  
 MASK:WFMDB:STATE, 2-215  
 MASK?, 2-197  
 MATH<x>:DEFine, 2-216  
 MATH<x>:FILTer:MODE, 2-217  
 MATH<x>:FILTer:RISetime, 2-217  
 MATH<x>:NUMavg, 2-218  
 MATH<x>:POSition, 2-218  
 MATH<x>:SCALE, 2-219  
 MATH<x>:UNIts?, 2-220  
 MATH<x>:WFMLabel, 2-220  
 MATH<x>?, 2-215  
 MEASUrement:ALL:VALue?, 2-222  
 MEASUrement:ANNOtations:STATE, 2-222  
 MEASUrement:LIST, 2-223  
 MEASUrement:LISTValue?, 2-224  
 MEASUrement:MEAS<x>:ALL?, 2-224  
 MEASUrement:MEAS<x>:EYEWindow, 2-225  
 MEASUrement:MEAS<x>:GATing:STATE, 2-226  
 MEASUrement:MEAS<x>:JITter, 2-227  
 MEASUrement:MEAS<x>:MAXimum?, 2-227  
 MEASUrement:MEAS<x>:MEAN?, 2-228  
 MEASUrement:MEAS<x>:MINimum?, 2-228  
 MEASUrement:MEAS<x>:NOISE, 2-229  
 MEASUrement:MEAS<x>:REFLevel<x>:ABSolute:  
   HIGH, 2-230  
 MEASUrement:MEAS<x>:REFLevel<x>:ABSolute:  
   LOW, 2-230  
 MEASUrement:MEAS<x>:REFLevel<x>:ABSolute:  
   MID, 2-231  
 MEASUrement:MEAS<x>:REFLevel<x>:  
   METHod, 2-232  
 MEASUrement:MEAS<x>:REFLevel<x>:RELative:  
   HIGH, 2-234  
 MEASUrement:MEAS<x>:REFLevel<x>:RELative:  
   LOW, 2-234  
 MEASUrement:MEAS<x>:REFLevel<x>:RELative:  
   MID, 2-235  
 MEASUrement:MEAS<x>:REFLevel<x>?, 2-229  
 MEASUrement:MEAS<x>:SETDefault, 2-236  
 MEASUrement:MEAS<x>:SOURce<x>:EDGE:  
   DIRection, 2-237  
 MEASUrement:MEAS<x>:SOURce<x>:EDGE:  
   SLOPe, 2-238  
 MEASUrement:MEAS<x>:SOURce<x>:  
   EDGE?, 2-236  
 MEASUrement:MEAS<x>:SOURce<x>:GATE<x>:  
   PCTPos, 2-240  
 MEASUrement:MEAS<x>:SOURce<x>:GATE<x>:  
   POS, 2-241  
 MEASUrement:MEAS<x>:SOURce<x>:  
   GATE<x>?, 2-239  
 MEASUrement:MEAS<x>:SOURce<x>:HILow:  
   METHod, 2-242  
 MEASUrement:MEAS<x>:SOURce<x>:HILow:  
   TRACK:HIGH:ENABle, 2-244  
 MEASUrement:MEAS<x>:SOURce<x>:HILow:  
   TRACK:HIGH:VALue, 2-245  
 MEASUrement:MEAS<x>:SOURce<x>:HILow:  
   TRACK:LOW:ENABle, 2-246  
 MEASUrement:MEAS<x>:SOURce<x>:HILow:  
   TRACK:LOW:VALue, 2-247  
 MEASUrement:MEAS<x>:SOURce<x>:  
   HILow?, 2-242  
 MEASUrement:MEAS<x>:SOURce<x>:WFM, 2-248  
 MEASUrement:MEAS<x>:SOURce<x>:WFMDB:  
   SIGType, 2-249  
 MEASUrement:MEAS<x>:SOURce<x>:WFMDB:  
   STATE, 2-250  
 MEASUrement:MEAS<x>:STATE, 2-251  
 MEASUrement:MEAS<x>:STATistics:CLEar, 2-252  
 MEASUrement:MEAS<x>:STDdev?, 2-252  
 MEASUrement:MEAS<x>:TYPE, 2-253  
 MEASUrement:MEAS<x>:UNIts?, 2-257  
 MEASUrement:MEAS<x>:VALue?, 2-258  
 MEASUrement:STATistics:ENABle, 2-259  
 MEASUrement:STATistics:WEIghting, 2-260  
 MEASUrement?, 2-221

## O

\*OPC, 2-260



\*OPT?, 2-261

## P

PHAseref:CH<x>:FREQuency, 2-262

PHAseref:CH<x>:RANge?, 2-263

PHAseref:CH<x>:STAtus?, 2-264

PHAseref:CHAR, 2-262

PHAseref:MODE, 2-266

PHAseref:SOUrce, 2-267

\*PSC, 2-268

## R

RECAIl:SETUp, 2-269

RECAIl:WAVEform, 2-270

REF<x>:POSItion, 2-270

REF<x>:SCAlE, 2-271

REF<x>:WFMLAbel, 2-271

\*RST, 2-272

## S

SAVe:SETUp, 2-273

SAVe:WAVEform, 2-273

SELEct:CH<x>, 2-275

SELEct:CONTRol, 2-275

SELEct:MATH<x>, 2-276

SELEct:REF<x>, 2-277

SELEct?, 2-274

SET?, 2-278

\*SRE, 2-279

\*STB?, 2-279

SYNC:TIMEOUT, 2-280

SYSTem:PROPErties:CH<x>:BANDwidth?, 2-281

SYSTem:PROPErties:CH<x>:CAPacitance?, 2-281

SYSTem:PROPErties:CH<x>:CLKRec?, 2-282

SYSTem:PROPErties:CH<x>:CLKUser?, 2-282

SYSTem:PROPErties:CH<x>:DYNamic?, 2-283

SYSTem:PROPErties:CH<x>:EXTender?, 2-283

SYSTem:PROPErties:CH<x>:FILTer?, 2-283

SYSTem:PROPErties:CH<x>:IMPedance?, 2-284

SYSTem:PROPErties:CH<x>:MODElnum?, 2-284

SYSTem:PROPErties:CH<x>:NONDestruct?, 2-284

SYSTem:PROPErties:CH<x>:OPERating?, 2-285

SYSTem:PROPErties:CH<x>:PRANge?, 2-285

SYSTem:PROPErties:CH<x>:PRObe:

DYNamic?, 2-286

SYSTem:PROPErties:CH<x>:PRObe:  
IMPedance?, 2-286

SYSTem:PROPErties:CH<x>:PRObe:  
MODElnum?, 2-287

SYSTem:PROPErties:CH<x>:PRObe:SCAlE?, 2-287

SYSTem:PROPErties:CH<x>:PRObe:  
SERialnum?, 2-287

SYSTem:PROPErties:CH<x>:RISetime?, 2-288

SYSTem:PROPErties:CH<x>:SERialnum?, 2-288

SYSTem:PROPErties:CH<x>:TEKPDriver?, 2-288

SYSTem:PROPErties:CH<x>:TEKPVersion?, 2-289

SYSTem:PROPErties:CH<x>:WLENgth?, 2-289

SYSTem:PROPErties:EFEHWver?, 2-289

SYSTem:PROPErties:EFESERialnum?, 2-290

SYSTem:PROPErties:GROUp<x>:

TEKPVersion?, 2-290

SYSTem:PROPErties:MAInframe:KEY?, 2-290

SYSTem:PROPErties:MAInframe:

MODElnum?, 2-291

SYSTem:PROPErties:MAInframe:SERialnum?, 2-291

SYSTem:PROPErties:MAInframe:SWVersion?, 2-291

SYSTem:PROPErties:MAInframe:UNIQueid?, 2-291

SYSTem:PROPErties:OFEHWver?, 2-292

SYSTem:PROPErties:OFESERialnum?, 2-292

SYSTem:PROPErties:ONTime?, 2-292

SYSTem:PROPErties:POWErups?, 2-293

SYSTem:PROPErties:PROCHWver?, 2-293

SYSTem:PROPErties:TBHWver?, 2-293

SYSTem:PROPErties:TBSErIalnum?, 2-294

SYSTem:PROPErties:TOTALontime?, 2-294

## T

TDR:CH<x>:DIFfpreset, 2-295

TDR:CH<x>:PRESET, 2-296

TDR:CH<x>:STEP:DESkew, 2-297

TDR:CH<x>:STEP:POLarity, 2-297

TDR:CH<x>:STEP:STATE, 2-298

TDR:CH<x>:UNIIts, 2-298

TDR:EXT10MHZref:FREQ, 2-299

TDR:INTRate, 2-300

TDR:REF10Mhz, 2-300

TDR?, 2-294

TIME, 2-301

TRIGger:AUTOSync, 2-302

TRIGger:AUTOSync:DATARate, 2-302

TRIGger:AUTOSync:DCRAtio, 2-303

TRIGger:AUTOSync:PLENgth, 2-304

TRIGger:CH<x>:CLKRec:LIST?, 2-304  
 TRIGger:CH<x>:CLKRec:RANge, 2-305  
 TRIGger:CH<x>:CLKRec:USER, 2-305  
 TRIGger:CH<x>:CLKRec:VALue, 2-306  
 TRIGger:CLKRec:CRC:AUTOSConfigdev, 2-309  
 TRIGger:CLKRec:CRC:CAPabilities?, 2-309  
 TRIGger:CLKRec:CRC:CLOCKAmplitude, 2-310  
 TRIGger:CLKRec:CRC:CLOCKOutput, 2-311  
 TRIGger:CLKRec:CRC:CREATESTandard, 2-311  
 TRIGger:CLKRec:CRC:DATARate?, 2-312  
 TRIGger:CLKRec:CRC:DELETETStandard, 2-312  
 TRIGger:CLKRec:CRC:DEVICEName, 2-313  
 TRIGger:CLKRec:CRC:EDGEDensity?, 2-313  
 TRIGger:CLKRec:CRC:  
     EDGEDENSITYMode, 2-314  
 TRIGger:CLKRec:CRC:EQualizer, 2-314  
 TRIGger:CLKRec:CRC:EXPrev?, 2-315  
 TRIGger:CLKRec:CRC:FPGarev?, 2-315  
 TRIGger:CLKRec:CRC:HALFRate?, 2-316  
 TRIGger:CLKRec:CRC:HWRev?, 2-316  
 TRIGger:CLKRec:CRC:LOCKCount?, 2-317  
 TRIGger:CLKRec:CRC:LOCKFp, 2-317  
 TRIGger:CLKRec:CRC:LOCKMode, 2-317  
 TRIGger:CLKRec:CRC:LOCKRange, 2-318  
 TRIGger:CLKRec:CRC:LOCKState?, 2-318  
 TRIGger:CLKRec:CRC:LOOPBandwidth, 2-319  
 TRIGger:CLKRec:CRC:MODElnum?, 2-319  
 TRIGger:CLKRec:CRC:NOMEDgedensity, 2-320  
 TRIGger:CLKRec:CRC:NOMFrequency, 2-320  
 TRIGger:CLKRec:CRC:PEAKing, 2-321  
 TRIGger:CLKRec:CRC:PHASEERRLimit, 2-321  
 TRIGger:CLKRec:CRC:PHASEERRP2P?, 2-322  
 TRIGger:CLKRec:CRC:PHASEERRRMS?, 2-322  
 TRIGger:CLKRec:CRC:RCONfigdevice, 2-323  
 TRIGger:CLKRec:CRC:RELock, 2-323  
 TRIGger:CLKRec:CRC:RESETLockcount, 2-324  
 TRIGger:CLKRec:CRC:SCONfigdevice, 2-324  
 TRIGger:CLKRec:CRC:SERIalnum?, 2-325  
 TRIGger:CLKRec:CRC:STANdard, 2-325  
 TRIGger:CLKRec:CRC:STANDARDList?, 2-326  
 TRIGger:CLKRec:CRC:SUBCLOCKAmpl, 2-326  
 TRIGger:CLKRec:CRC:SUBCLOCKDiv, 2-327  
 TRIGger:CLKRec:CRC:  
     SUBCLOCKDIVList?, 2-327  
 TRIGger:CLKRec:CRC:SUBCLOCKOutput, 2-328  
 TRIGger:CLKRec:CRC:SWRev?, 2-328  
 TRIGger:CLKRec:SOUrce, 2-329

TRIGger:HIFreq, 2-329  
 TRIGger:HOLDoff, 2-330  
 TRIGger:INTRate, 2-331  
 TRIGger:LEVel, 2-332  
 TRIGger:MODE, 2-333  
 TRIGger:PSYNc:DATARate, 2-334  
 TRIGger:PSYNc:DCRAtio, 2-334  
 TRIGger:PSYNc:PLENght, 2-335  
 TRIGger:SETLevel, 2-336  
 TRIGger:SLOpe, 2-336  
 TRIGger:SOUrce, 2-337  
 TRIGger:STATE?, 2-338  
 TRIGger?, 2-301

## U

UNLock, 2-339

## V

VERBose, 2-340

## W

\*WAI, 2-340  
 WAVFrm?, 2-341  
 WFMDB:DISPlay:EMPHasis, 2-342  
 WFMDB:DISPlay:GRADing, 2-343  
 WFMDB:DISPlay:GRADMethod, 2-344  
 WFMDB:DISPlay:INVert, 2-345  
 WFMDB:DISPlay?, 2-342  
 WFMDB:WFMDB<x>:CLEar, 2-346  
 WFMDB:WFMDB<x>:DISplay, 2-346  
 WFMDB:WFMDB<x>:ENABLE, 2-347  
 WFMDB:WFMDB<x>:PERSistence:COUNT, 2-347  
 WFMDB:WFMDB<x>:PERSistence:MODE, 2-348  
 WFMDB:WFMDB<x>:PERSistence:  
     SAMPles?, 2-349  
 WFMDB:WFMDB<x>:SOUrce, 2-350  
 WFMDB:WFMDB<x>?, 2-345  
 WFMDB?, 2-341  
 WFMInpre:BIT\_Nr?, 2-351  
 WFMInpre:BN\_Fmt, 2-351  
 WFMInpre:BYT\_Nr?, 2-352  
 WFMInpre:BYT\_Or, 2-352  
 WFMInpre:ENCdg, 2-353  
 WFMInpre:NR\_Pt, 2-353  
 WFMInpre:PT\_Fmt, 2-354  
 WFMInpre:WFMLLabel, 2-355

---

WFMInpre:XINcr, 2-356  
WFMInpre:XMUlt, 2-356  
WFMInpre:XOFF, 2-357  
WFMInpre:XUNit, 2-357  
WFMInpre:XZEro, 2-358  
WFMInpre:YMUlt, 2-358  
WFMInpre:YOFF, 2-359  
WFMInpre:YSCALE, 2-359  
WFMInpre:YUNit, 2-360  
WFMInpre:YZEro, 2-360  
WFMInpre?, 2-350  
WFMOutpre:BIT\_Nr?, 2-361  
WFMOutpre:BN\_Fmt, 2-362  
WFMOutpre:BYT\_Nr?, 2-362  
WFMOutpre:BYT\_Or, 2-363  
WFMOutpre:ENCdg, 2-363  
WFMOutpre:NR\_Pt?, 2-364  
WFMOutpre:PT\_Fmt?, 2-364  
WFMOutpre:WFId?, 2-365  
WFMOutpre:WFMLabel?, 2-365  
WFMOutpre:XINcr?, 2-365  
WFMOutpre:XMUlt?, 2-366  
WFMOutpre:XOFF?, 2-366  
WFMOutpre:XUNit?, 2-367  
WFMOutpre:XZEro?, 2-367  
WFMOutpre:YMUlt?, 2-367  
WFMOutpre:YOFF?, 2-368  
WFMOutpre:YSCALE?, 2-368  
WFMOutpre:YUNit?, 2-369  
WFMOutpre:YZEro?, 2-369  
WFMOutpre?, 2-361