RF Generic Signals
Application Plug-in
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.

SourceXpress® is a registered 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.tek.com to find contacts in your area.
Table of Contents

Getting Started

Introduction ....................................................................................................... 1-1
Documentation.................................................................................................... 1-2

Syntax and Commands

Command Syntax................................................................................................. 2-1
Command Groups ................................................................................................ 2-3

Control commands........................................................................................... 2-3
Compile commands ......................................................................................... 2-3
Carrier setup commands ................................................................................... 2-5
Digital modulation commands ........................................................................... 2-5
Analog modulation commands ......................................................................... 2-6
Custom modulation commands ....................................................................... 2-7
Data commands ............................................................................................... 2-8
Filter commands ............................................................................................. 2-8
Hopping commands ......................................................................................... 2-9
IQ impairments commands ............................................................................ 2-10
Power ramp commands ................................................................................... 2-11
Interference addition commands ................................................................... 2-12
Distortion commands ..................................................................................... 2-13
Multipath commands ..................................................................................... 2-13
S-Parameter commands ................................................................................... 2-14
Sub Carrier Modulation commands ................................................................ 2-16
Commands in alphabetical order...................................................................... 2-17
Introduction

This programmer manual provides information on how to use commands for remotely controlling the RF Generic Signals plug-in application.

The RF Generic Waveform plug-in can be installed in the SourceXpress software application, AWG70000 series instruments, and AWG5200 series instruments.

Communication with the plug-in is through the programmatic interface of these products. Using a single VISA or raw socket session, it is possible to communicate with SourceXpress, AWG70000 series instruments, and AWG5200 series instruments.

For information on the Remote Control, GPIB Parameters, LAN Parameters, Connecting to the Instrument using GPIB, and Setting up GPIB Communication, refer to the *AWG70000 series Arbitrary Waveform Generators Programmer Manual* or the *AWG5200 series Arbitrary Waveform Generators Programmer Manual*. 
In addition to this RF Generic Signals Programmer Guide, the following documentation is included with this application:

- **RF Generic Signals Help.** The help provides in-depth operation and user interface help.

- **RF Generic Signals User Manual (PDF).** This is adapted from the RF Generic Signals help system.
Command Syntax

For information on the Syntax Overview, Command and Query Structure, Clearing the Instrument, Command Entry, Parameter Types, SCPI Commands and Queries, refer to the *AWG70000 series Arbitrary Waveform Generators Programmer Manual* or the *AWG5200 series Arbitrary Waveform Generators Programmer Manual*. 
Command Groups

Control commands

Table 2-1: Control commands and their descriptions

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>WPLugin:ACTive</td>
<td>Sets or returns the active waveform plug-in.</td>
</tr>
<tr>
<td>RFGSignal:RESet</td>
<td>Resets the RF Generic Signal plug-in by setting all the values within the module to their default values.</td>
</tr>
</tbody>
</table>

Compile commands

Table 2-2: Compile commands and their descriptions

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFGSignal:COMPile</td>
<td>Compiles and generates a waveform using the RF Generic Signal plug-in compile settings.</td>
</tr>
<tr>
<td>RFGSignal:COMPile:AWARound</td>
<td>Sets or returns the Adjust Wrap Around state (enabled or disabled) for the Compile Settings.</td>
</tr>
<tr>
<td>RFGSignal:COMPile:CANCel</td>
<td>Cancels a compilation currently in progress.</td>
</tr>
<tr>
<td>RFGSignal:COMPile:CASSign</td>
<td>Sets or returns the state (enabled or disabled) to compile the waveform and immediately assign it to a specified channel (enabled) or just compile the waveform (disabled).</td>
</tr>
<tr>
<td>RFGSignal:COMPile:CHANnel:I</td>
<td>Sets or returns which channel the I signal is assigned to upon compile when the signal format is set to IQ.</td>
</tr>
<tr>
<td>RFGSignal:COMPile:CHANnel:IQ</td>
<td>Sets or returns the playout channel intended for the compiled IQ signal after up-conversion using the internal IQ modulator.</td>
</tr>
<tr>
<td>RFGSignal:COMPile:CHANnel:Q</td>
<td>Sets or returns which channel the Q signal is assigned to upon compile when the signal format is set to IQ.</td>
</tr>
<tr>
<td>RFGSignal:COMPile:CHANnel:RF</td>
<td>Sets or returns which channel the RF signal is assigned to upon compile when the signal format is set to RF/IF.</td>
</tr>
<tr>
<td>RFGSignal:COMPile:CORRection:APPLy</td>
<td>Sets or returns the Apply Corrections File state (enabled or disabled) for the Compile Settings.</td>
</tr>
<tr>
<td>RFGSignal:COMPile:CORRection:PATH</td>
<td>Sets or returns the RF Correction filename and filepath to use when compiling an RF signal.</td>
</tr>
<tr>
<td>RFGSignal:COMPile:CORRection:PATH:I</td>
<td>Sets or returns the I Correction filename and filepath to use when compiling and IQ signal.</td>
</tr>
<tr>
<td>RFGSignal:COMPile:CORRection:PATH:IQ</td>
<td>Sets or returns the IQ Correction filename and filepath to use when compiling and IQ signal.</td>
</tr>
<tr>
<td>RFGSignal:COMPile:CORRection:PATH:Q</td>
<td>Sets or returns the Q Correction filename and filepath to use when compiling and IQ signal.</td>
</tr>
<tr>
<td>RFGSignal:COMPile:CORRection:TYPE</td>
<td>Sets or returns the type of IQ correction file (a single IQ file or individual I and Q files) to apply when compiling.</td>
</tr>
</tbody>
</table>
Table 2-2: Compile commands and their descriptions (cont.)

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFGSignal:COMPile:DUPConverter</td>
<td>Sets or returns the Internal IQ Modulator state (enabled or disabled) when compiling.</td>
</tr>
<tr>
<td>RFGSignal:COMPile:FDRange</td>
<td>Sets or returns the Fit to full dynamic range state (enabled or disabled) for the Compile Settings.</td>
</tr>
<tr>
<td>RFGSignal:COMPile:ICHannel</td>
<td><strong>NOTE.</strong> This command exists for backwards compatibility. Use the command RFGSignal:COMPile:CHANnel:I.</td>
</tr>
<tr>
<td></td>
<td>Sets or returns which channel the I signal is assigned to upon compile when the signal format is set to IQ.</td>
</tr>
<tr>
<td>RFGSignal:COMPile:MARKer:ENABle</td>
<td>Sets or returns the Marker Data state (enabled or disabled) for the Compile Settings.</td>
</tr>
<tr>
<td>RFGSignal:COMPile:MARKer[n]:CFRequency</td>
<td>Sets or returns the Marker Data clock frequency when the Marker Data type is set to Clock Frequency.</td>
</tr>
<tr>
<td>RFGSignal:COMPile:MARKer[n]:MDATa</td>
<td>Sets or returns the Marker Data Type of the specified marker.</td>
</tr>
<tr>
<td>RFGSignal:COMPile:NAME</td>
<td>Sets or returns the name of the compiled waveform.</td>
</tr>
<tr>
<td>RFGSignal:COMPile:OSAMpling</td>
<td>Sets or returns the over sampling rate used to determine the sampling rate of the compiled signal.</td>
</tr>
<tr>
<td>RFGSignal:COMPile:PLAY</td>
<td>Sets or returns the Play after assign state (enabled or disabled) for the Compile Settings.</td>
</tr>
<tr>
<td>RFGSignal:COMPile:QCHannel</td>
<td><strong>NOTE.</strong> This command exists for backwards compatibility. Use the command RFGSignal:COMPile:CHANnel:Q.</td>
</tr>
<tr>
<td></td>
<td>Sets or returns which channel the Q signal is assigned to upon compile when the signal format is set to IQ.</td>
</tr>
<tr>
<td>RFGSignal:COMPile:RFCChannel</td>
<td><strong>NOTE.</strong> This command exists for backwards compatibility. Use the command RFGSignal:COMPile:CHANnel:RF.</td>
</tr>
<tr>
<td></td>
<td>Sets or returns which channel the RF waveform is assigned to upon compile when the signal format is set to RF.</td>
</tr>
<tr>
<td>RFGSignal:COMPile:SFORmat</td>
<td>Sets or returns the signal format of the signal to be compiled.</td>
</tr>
<tr>
<td>RFGSignal:COMPile:SRATe</td>
<td>Sets or returns the sampling rate for the compile settings.</td>
</tr>
<tr>
<td>RFGSignal:COMPile:SRATe:AUTO</td>
<td>Sets or returns the value that indicates if the sampling rate will be automatically calculated at compile time.</td>
</tr>
<tr>
<td>RFGSignal:COMPile:WLENgth</td>
<td>Sets or returns the waveform length. If the waveform length is not set to auto then the value for waveform length is based on the value provided.</td>
</tr>
<tr>
<td>RFGSignal:COMPile:WLENgth:AUTO</td>
<td>Sets or returns if the waveform length will be automatically calculated at compile time.</td>
</tr>
</tbody>
</table>
Table 2-2: Compile commands and their descriptions (cont.)

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFGSignal:COMPile:WLENgh:TYPE</td>
<td>Sets or returns the waveform length unit type that is used to set the waveform length. The waveform length can be provided in terms of samples, symbols, or seconds</td>
</tr>
<tr>
<td>RFGSignal:COMPile:WOverwrite</td>
<td>Sets or returns the state (enabled or disabled) wether or not to overwrite the existing waveform if the waveform already exists in the Waveform list.</td>
</tr>
</tbody>
</table>

Carrier setup commands

Table 2-3: Carrier setup commands and their descriptions

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFGSignal:CARRier:ADD</td>
<td>Creates the specified number of carriers in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier:COUNt?</td>
<td>Returns the carrier count in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:Carrier[n]:BBOFfset</td>
<td>Sets or returns the base band Offset for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:DELeTe</td>
<td>Deletes the specified carrier from the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:FREQuency</td>
<td>Sets or returns the Carrier Frequency for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:Carrier[n]:IQAMplitude</td>
<td>Sets or returns the IQ Amplitude for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:NOISe:BANDwidth</td>
<td>Sets or returns the Bandwidth for the Noise Carrier for the specified carrier in the carrier table when the Carrier Type is set to Noise.</td>
</tr>
<tr>
<td>RFGSignal:Carrier[n]:PHASe</td>
<td>Sets or returns the Phase for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:RFAMplitude</td>
<td>Sets or returns the Amplitude for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:TYPE</td>
<td>Sets or returns the Carrier Type for the specified carrier in the carrier table.</td>
</tr>
</tbody>
</table>

Digital modulation commands

Table 2-4: Digital modulation commands and their descriptions

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFGSignal:CARRier[n]:DMODulation:APSK</td>
<td>Sets or returns the APSK Modulation type for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:DMODulation:APSK:PROTation</td>
<td>Sets or returns the Phase Rotation of the selected APSK Ring Index for the specified carrier in the carrier table.</td>
</tr>
</tbody>
</table>
### Command Groups

#### Digital modulation commands and their descriptions (cont.)

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFGSignal:CARRier[n]:DMODulation:APSK:RADius</td>
<td>Sets or returns the Radius of the selected APSK Ring Index for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:DMODulation:APSK:SRINg</td>
<td>Sets or returns the selected APSK Ring Index for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:DMODulation:APSK:SYMBols</td>
<td>Sets or returns the Number of Symbols of the selected APSK Ring Index for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:DMODulation:ASK:MINDex</td>
<td>Sets or returns the ASK Mod Index for ASK modulation for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:DMODulation:CPM:MINDex</td>
<td>Sets or returns the Index for CPM modulation for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:DMODulation:FSK</td>
<td>Sets or returns the FSK Modulation type for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:DMODulation:FSK:PDEViation</td>
<td>Sets or returns the FSK Peak Deviation for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:DMODulation:NDPSK:INDex</td>
<td>Sets or returns the ‘n’ value (the kind of DPSK) for nDPSK modulation for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:DMODulation:NDPSK:PROTation</td>
<td>Sets or returns the Phase Rotation of nDPSK modulation for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:DMODulation:PSK</td>
<td>Sets or returns the PSK modulation type for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:DMODulation:QAM</td>
<td>Sets or returns the QAM Modulation type for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:DMODulation:SRATe</td>
<td>Sets or returns the Digital Modulation Symbol Rate for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:DMODulation:TYPe</td>
<td>Sets or returns the digital modulation type for the specified carrier in the carrier table.</td>
</tr>
</tbody>
</table>

#### Analog modulation commands

**Table 2-5: Analog modulation commands and their descriptions**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFGSignal:CARRier[n]:AMODulation:AMModulation:AMINdex</td>
<td>Creates the specified number of carriers in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:AMODulation:FMModulation:FDEViation</td>
<td>Sets or returns the Frequency Deviation for the FM Analog Modulation for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:AMODulation:MFRrequency</td>
<td>Sets or returns the Modulating Frequency of the sinusoidal, triangular, or square Modulating Signals for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:AMODulation:MSIGnal:FNAMe</td>
<td>Sets or returns the file name for User Defined Modulating Signal for the specified carrier in the carrier table.</td>
</tr>
</tbody>
</table>
### Table 2-5: Analog modulation commands and their descriptions (cont.)

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFGSignal:CARRier[n]:AMODulation:MSIGNAL:INTERpolation</td>
<td>Sets or returns the Interpolation type for User Defined Modulating Signal for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:AMODulation:MSIGNAL:POFFset</td>
<td>Sets or returns the Phase Offset of the sinusoidal, triangular, or square Modulating Signals for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:AMODulation:MSIGNAL:SRATE</td>
<td>Sets or returns the Sampling Rate for User Defined Modulating Signal for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:AMODulation:MSIGNAL:TYPE</td>
<td>Sets or returns the Modulating Signal type for Analog Modulation for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:AMODulation:PMODulation:PDEViation</td>
<td>Sets or returns the Phase Deviation for the PM Analog Modulation for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:AMODulation:TYPE</td>
<td>Sets or returns the Analog Modulation type for the specified carrier in the carrier table.</td>
</tr>
</tbody>
</table>

### Custom modulation commands

#### Table 2-6: Custom modulation commands and their descriptions

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFGSignal:CARRier[n]:CMODulation:ADDMAP</td>
<td>Adds the specified number of maps to the Custom Modulation table for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:CMODulation:DELMAP</td>
<td>Deletes a single Custom Modulation map for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:CMODulation:ISYMBOl</td>
<td>Sets or returns the I symbol value of the currently selected Custom Modulation map for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:CMODulation:MMODE</td>
<td>Sets or returns the Modulation mode for the Custom Modulation for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:CMODulation:MOFFSET</td>
<td>Sets or returns the Offset modulation state (enabled or disabled) for Custom Modulation for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:CMODulation:QSYMBOl</td>
<td>Sets or returns the Q symbol value of the currently selected Custom Modulation map for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:CMODulation:SELMAP</td>
<td>Sets or returns the selected map in the Custom Modulation table for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:CMODulation:SRATE</td>
<td>Sets or returns the Symbol Rate for the Custom Modulation for the specified carrier in the carrier table.</td>
</tr>
</tbody>
</table>
## Data commands

Table 2-7: Data commands and their descriptions

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFGSignal:CARRier[n]:DATA</td>
<td>Sets or returns the Data Pattern type for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:DATA:CODing</td>
<td>Sets or returns the coding type for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:DATA:FILE</td>
<td>Sets or returns the pattern file and file path for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:DATA:PATTern</td>
<td>Sets or returns the user defined data pattern for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:DATA:PRBS</td>
<td>Sets or returns the PBRS data type to one of the selected PRBS types for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:DATA:PRBS:UDEFined:POLYnomial</td>
<td>Sets or returns the PRBS Polynomial Expression for the User Defined PRBS for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:DATA:PRBS:UDEFined:RESet</td>
<td>Resets the User Defined PRBS polynomial expression and shift register to their default values for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:DATA:PRBS:UDEFined:SREGister</td>
<td>Sets or returns the PRBS Shift register initial value for the User Defined PRBS for the specified carrier in the carrier table.</td>
</tr>
</tbody>
</table>

## Filter commands

Table 2-8: Filter commands and their descriptions

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFGSignal:CARRier[n]:FILTer:ALPHa</td>
<td>Sets or returns the alpha (filter roll off) value for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:FILTer:CLENgth</td>
<td>Sets or returns the Filter Convolution Length in symbols for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:FILTer:FILe</td>
<td>Sets or returns the path of the filter file for user defined filter for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:FILTer:TYPe</td>
<td>Sets or returns the Filter Type for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:FILTer:WINDow:CRIPple</td>
<td>Sets or returns the Chebyshev Ripple value when the Filter Window type is Chebyshev Ripple for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:FILTer:WINDow:KAISer</td>
<td>Sets or returns the Kaiser Parameter when the Filter Window type is Kaiser for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:FILTer:WINDow:TYPE</td>
<td>Sets or returns the Filter Window type for the specified carrier in the carrier table.</td>
</tr>
</tbody>
</table>
### Hopping commands

#### Table 2-9: Hopping commands and their descriptions

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFGSignal:CARRier[n]:HOPPing:CUSTom:RLISt</td>
<td>Sets or returns the Repeat List state (enabled or disabled) for the Frequency Hop List for the specified carrier in the carrier table. When enabled, the Frequency Hop List repeats when the Hop List has been completed.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:HOPPing:LIST:ADD</td>
<td>Adds a single entry to the end of the Frequency Hop List or the Frequency Avoid List for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:HOPPing:LIST:AOFFset</td>
<td>Sets or returns the Amplitude Offset of the currently selected hop for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:HOPPing:LIST:COUN?</td>
<td>Returns the number of hops in the Hop List or the number of entries in the Frequency Avoid List for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:HOPPing:LIST:DELeTe</td>
<td>Removes all entries within the Frequency Hop List or the Frequency Avoid List for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:HOPPing:LIST:FOFFset</td>
<td>Sets or returns the Frequency Offset (Relative Frequency) of the currently selected hop for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:HOPPing:LIST:FREQuency</td>
<td>Sets or returns the Frequency of the currently selected hop in the Frequency Hop List for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:HOPPing:LIST:HDURation</td>
<td>Sets or returns the Hop Duration of the currently selected hop for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:HOPPing:LIST:INSert</td>
<td>Inserts a single entry within the Frequency Hop List or the Frequency Avoid List for the specified carrier in the carrier table. The selected entry point must already exist in the table, otherwise an error is returned.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:HOPPing:LIST:SELect</td>
<td>Sets or returns the selected Hop number in the Frequency Hop List or the Index number in the Frequency Avoid List for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:HOPPing:LIST:SSINdex</td>
<td>Sets or returns the Symbol Start Index (Symbol Index) of the currently selected hop for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:HOPPing:PATTern</td>
<td>Sets or returns the Hopping Pattern for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:HOPPing:PRBS</td>
<td>Sets or returns the Hopping PRBS pattern for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:HOPPing:PRBS:UDEFined:POLYnomial</td>
<td>Sets or returns the Hopping PRBS Polynomial Expression for the User Defined PRBS for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:HOPPing:PRBS:UDEFined:RESet</td>
<td>Resets the Hopping User Defined PRBS type to its default value for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:HOPPing:PRBS:UDEFined:SREGister</td>
<td>Sets or returns the Hopping PRBS Polynomial Expression for the User Defined PRBS for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:HOPPing:RANGe:ALISt[:ENABle]</td>
<td>Sets or returns the Frequency Avoid List state (enabled or disabled) for the Pseudo Random Range Hopping Pattern for the specified carrier in the carrier table.</td>
</tr>
</tbody>
</table>
### Command Groups

**Table 2-9: Hopping commands and their descriptions (cont.)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFGSignal:CARRier[n]:HOPPing:RANGe[:FREQuency]:MAXimum</td>
<td>Sets or returns the Maximum Frequency value for the Pseudo Random Range Hopping Pattern for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:HOPPing:RANGe[:FREQuency]:MINimum</td>
<td>Sets or returns the Minimum Frequency value for the Pseudo Random Range Hopping Pattern for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:HOPPing:RANGe[:FREQuency]:SPACing</td>
<td>Sets or returns the Frequency Spacing value for the Pseudo Random Range Hopping Pattern for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:HOPPing:TIME</td>
<td>Sets or returns the Hop Time type for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:HOPPing:TIME:HPSecond</td>
<td>Sets or returns the Hops per Second for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:HOPPing:TIME:SPHop</td>
<td>Sets or returns the Symbols per Hop for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:HOPPing:TON</td>
<td>Sets or returns the Hopping state (enabled or disabled) for the specified carrier in the carrier table.</td>
</tr>
</tbody>
</table>

### IQ impairments commands

**Table 2-10: IQ impairments commands and their descriptions**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFGSignal:CARRier[n]:IQIMpairment:CLEAkage:IOFFset</td>
<td>Sets or returns the I Offset percentage for Carrier Leakage for IQ Impairments for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:IQIMpairment:CLEAkage:IQOFFset</td>
<td>Sets or returns the IQ Offset level for Carrier Leakage for IQ Impairments for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:IQIMpairment:CLEAkage:QOFFset</td>
<td>Sets or returns the Q Offset percentage for Carrier Leakage for IQ Impairments for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:IQIMpairment:CLEAkage:TON</td>
<td>Sets or returns the Carrier Leakage state (enabled or disabled) for IQ Impairments for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:IQIMpairment:IQIMbalance:IMBalance</td>
<td>Sets or returns the Imbalance percentage for IQ Imbalance for IQ Impairments for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:IQIMpairment:IQIMbalance:TON</td>
<td>Sets or returns the IQ Imbalance state (enabled or disabled) for IQ Impairments for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:IQIMpairment:IOSWap:TON</td>
<td>Sets or returns the Swap I &amp; Q state (enabled or disabled) for IQ Impairments for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:IQIMpairment:NLDistortion:AM2K</td>
<td>Sets or returns the k2 level for AM/AM Nonlinear Distortions for IQ Impairments for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:IQIMpairment:NLDistortion:AM3K</td>
<td>Sets or returns the k3 level for AM/AM Nonlinear Distortions for IQ Impairments for the specified carrier in the carrier table.</td>
</tr>
</tbody>
</table>
Table 2-10: IQ impairments commands and their descriptions (cont.)

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>RFGSignal:CARRier[n]:IQIMpairment:NLDistortion:PM2K</code></td>
<td>Sets or returns the k2 level for AM/PM Nonlinear Distortions for IQ Impairments for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td><code>RFGSignal:CARRier[n]:IQIMpairment:NLDistortion:PM3K</code></td>
<td>Sets or returns the k3 level for AM/PM Nonlinear Distortions for IQ Impairments for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td><code>RFGSignal:CARRier[n]:IQIMpairment:NLDistortion:TON</code></td>
<td>Sets or returns the Nonlinear Distortions state (enabled or disabled) for IQ Impairments for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td><code>RFGSignal:CARRier[n]:IQIMpairment:QERRor:ERRor</code></td>
<td>Sets or returns the I/Q Error percentage for Quadrature Error for IQ Impairments for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td><code>RFGSignal:CARRier[n]:IQIMpairment:QERRor:TON</code></td>
<td>Sets or returns the Quadrature Error state (enabled or disabled) for IQ Impairments for the specified carrier in the carrier table.</td>
</tr>
</tbody>
</table>

Power ramp commands

Table 2-11: Power ramp commands and their descriptions

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>RFGSignal:CARRier[n]:PRAMp:DTYPe</code></td>
<td>Sets or returns the Duration Units used for the Power Ramp table for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td><code>RFGSignal:CARRier[n]:PRAMp:DURation</code></td>
<td>Sets or returns the Power Ramp Duration for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td><code>RFGSignal:CARRier[n]:PRAMp:FUNCtion</code></td>
<td>Sets or returns the Power Ramp Function for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td><code>RFGSignal:CARRier[n]:PRAMp:ILEVel</code></td>
<td>Sets or returns the Power Ramp Initial Level of the currently selected hop for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td><code>RFGSignal:CARRier[n]:PRAMp:LEVel:ADD</code></td>
<td>Adds new entries to the Power Ramp table for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td><code>RFGSignal:CARRier[n]:PRAMp:LEVel:DELeete</code></td>
<td>Deletes entries from the Power Ramp table for the specified carrier in the carrier table. An error is returned if the entry does not exist.</td>
</tr>
<tr>
<td><code>RFGSignal:CARRier[n]:PRAMp:LEVel:DURation</code></td>
<td>Sets or returns the Duration of the currently selected Power Ramp table index for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td><code>RFGSignal:CARRier[n]:PRAMp:LEVel:ESYMbol</code></td>
<td>Sets or returns the End Symbol of the currently selected Power Ramp table index for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td><code>RFGSignal:CARRier[n]:PRAMp:LEVel:POWer</code></td>
<td>Sets or returns the Power Level of the currently selected Power Ramp table index for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td><code>RFGSignal:CARRier[n]:PRAMp:LEVel:SELeect</code></td>
<td>Sets or returns the selected index (row) of the Power Ramp table for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td><code>RFGSignal:CARRier[n]:PRAMp:LEVel:SSYMbol</code></td>
<td>Sets or returns the Start Symbol of the currently selected Power Ramp table index for the specified carrier in the carrier table.</td>
</tr>
</tbody>
</table>
Command Groups

Table 2-11: Power ramp commands and their descriptions (cont.)

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFGSignal:CARRier[n]:PRAMp:PEXTend</td>
<td>Sets or returns the Periodic Extend Power Levels state (enabled or disabled) for the Power Ramp table for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:PRAMp:TON</td>
<td>Sets or returns the Power Ramp state (enabled or disabled) for the specified carrier in the carrier table.</td>
</tr>
</tbody>
</table>

Interference addition commands

Table 2-12: Interference addition commands and their descriptions

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFGSignal:CARRier[n]:INTerference:ANOIse:BANDwidth</td>
<td>Sets or returns the Bandwidth for Additive Noise for Interference Addition for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:INTerference:ANOIse:EBNo</td>
<td>Sets or returns the Eb/No (bit energy per noise power) for Additive Noise for Interference Addition for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:INTerference:ANOIse:SNR</td>
<td>Sets or returns the SNR for Additive Noise for Interference Addition for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:INTerference:ANOIse:TON</td>
<td>Sets or returns the Additive Noise state (enabled or disabled) for Interference Addition for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:INTerference:FOFFset</td>
<td>Sets or returns the Frequency Offset frequency for Interference Addition for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:INTerference:FOFFset:TON</td>
<td>Sets or returns the Frequency Offset state (enabled or disabled) for Interference Addition for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:INTerference:SINusoidal:CI</td>
<td>Sets or returns the C/I level for Sinusoidal Interference for Interference Addition for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:INTerference:SINusoidal:COFFset</td>
<td>Sets or returns the Offset from Carrier frequency for Sinusoidal Interference for Interference Addition for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:CARRier[n]:INTerference:SINusoidal:TON</td>
<td>Sets or returns the Sinusoidal Interference state (enabled or disabled) for Interference Addition for the specified carrier in the carrier table.</td>
</tr>
</tbody>
</table>
## Distortion commands

Table 2-13: Distortion commands and their descriptions

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>RFGSignal:CARRier[n]:DISTortion:AMPLifier:AM3K</code></td>
<td>Sets or returns the k3 level for AM/AM Distortions for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td><code>RFGSignal:CARRier[n]:DISTortion:AMPLifier:AM5K</code></td>
<td>Sets or returns the k5 level for AM/AM Distortions for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td><code>RFGSignal:CARRier[n]:DISTortion:AMPLifier:LLEVEL</code></td>
<td>Sets or returns the Limiting Level value of amplifier distortion for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td><code>RFGSignal:CARRier[n]:DISTortion:AMPLifier:OPONt</code></td>
<td>Sets or returns the Operating Point value of amplifier distortion for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td><code>RFGSignal:CARRier[n]:DISTortion:AMPLifier:PM3K</code></td>
<td>Sets or returns the k3 level for AM/PM Distortions for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td><code>RFGSignal:CARRier[n]:DISTortion:AMPLifier:PM5K</code></td>
<td>Sets or returns the k5 level for AM/PM Distortions for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td><code>RFGSignal:CARRier[n]:DISTortion:AMPLifier:TON</code></td>
<td>Sets or returns the Distortion state (enabled or disabled) for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td><code>RFGSignal:CARRier[n]:DISTortion:AMPLifier:TYPe</code></td>
<td>Sets or returns the amplifier Distortion type for the specified carrier in the carrier table.</td>
</tr>
</tbody>
</table>

## Multipath commands

Table 2-14: Multipath commands and their descriptions

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>RFGSignal:CARRier[n]:MULTipath:ADDPath</code></td>
<td>Adds the specified number of paths in the Multipath table for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td><code>RFGSignal:CARRier[n]:MULTipath:AMPLitude</code></td>
<td>Sets or returns the Multipath Amplitude of the currently selected path for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td><code>RFGSignal:CARRier[n]:MULTipath:DELay</code></td>
<td>Sets or returns the Multipath Delay in symbols of the currently selected path for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td><code>RFGSignal:CARRier[n]:MULTipath:DELPath</code></td>
<td>Deletes a single path in the Multipath table for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td><code>RFGSignal:CARRier[n]:MULTipath:PHASE</code></td>
<td>Sets or returns the Multipath Phase of the currently selected path for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td><code>RFGSignal:CARRier[n]:MULTipath:SELPath</code></td>
<td>Sets or returns the selected MultiPath table index (row) for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td><code>RFGSignal:CARRier[n]:MULTipath:TURNon</code></td>
<td>Sets or returns the Multipath state (enabled or disabled) for the specified carrier in the carrier table.</td>
</tr>
</tbody>
</table>
## S-Parameter commands

Table 2-15: S-Parameter commands and their descriptions

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFGSN:SPAR:SFORmat</td>
<td>Sets or returns the currently used signal format for all S-Parameter values.</td>
</tr>
<tr>
<td>RFGSN:SPAR:SFORmat:LIQ</td>
<td>Sets or returns the Couple Settings (I,Q) state. When enabled, all I and Q S-Parameters are linked together (chained) so that all parameters match between I and Q.</td>
</tr>
<tr>
<td>RFGSN:SPAR:TON</td>
<td>Sets or returns the S-Parameter state (enabled or disabled).</td>
</tr>
<tr>
<td>RFGSN:SPAR:BANDwidth</td>
<td>Sets or returns the S-Parameter bandwidth when setting manually.</td>
</tr>
<tr>
<td>RFGSN:SPAR:BANDwidth:AUTO</td>
<td>Sets or returns the S-Parameter automatic bandwidth calculation setting.</td>
</tr>
<tr>
<td>RFGSN:SPAR:CASC:AGGR2[:ENABle]</td>
<td>Sets or returns the aggressor 2 signal type state (enabled or disabled) in Cascading mode. Aggressor2 signals are available when the number of ports is set to 12.</td>
</tr>
<tr>
<td>RFGSN:SPAR:CASC:AGGR[n]:AMPLitude</td>
<td>Sets or returns the specified Aggressor's amplitude, in Cascading mode.</td>
</tr>
<tr>
<td>RFGSN:SPAR:CASC:AGGR[n]:CTALk</td>
<td>Sets or returns the specified Aggressor's crosstalk type, in Cascading mode.</td>
</tr>
<tr>
<td>RFGSN:SPAR:CASC:AGGR[n]:DRATe</td>
<td>Sets or returns the specified Aggressor's data rate, in Cascading mode.</td>
</tr>
<tr>
<td>RFGSN:SPAR:CASC:AGGR[n]:SIGNal</td>
<td>Sets or returns specified Aggressor's signal type, in Cascading mode.</td>
</tr>
<tr>
<td>RFGSN:SPAR:CASC:AGGR[n]:SIGNal:FILE</td>
<td>Sets or returns the filepath to the aggressor file for the specified Aggressor, in Cascading mode.</td>
</tr>
<tr>
<td>RFGSN:SPAR:CASC:AGGR[n]:SIGNal:PRBS</td>
<td>Sets or returns the specified Aggressor's PRBS signal type, in Cascading mode.</td>
</tr>
<tr>
<td>RFGSN:SPAR:CASC:DEEMbed</td>
<td>Sets or returns whether the Cascading S-Parameters is to de-embed (invert) the S-Parameters, in Cascading mode.</td>
</tr>
<tr>
<td>RFGSN:SPAR:CASC:STAG[m]:DRX[n]</td>
<td>Sets or returns the S-Parameter port assignment of the specified Stage and the channel’s specified receiver port number (Rx-Port) in Cascading mode and Differential Signalling Scheme (where applicable).</td>
</tr>
<tr>
<td>RFGSN:SPAR:CASC:STAG[m]:DTX[n]</td>
<td>Sets or returns the S-Parameter port assignment of the specified Stage and the channel’s specified transmission port number (Tx-Port) in Cascading mode and Differential Signalling Scheme (where applicable).</td>
</tr>
<tr>
<td>RFGSN:SPAR:CASC:STAG[m]:ENABLE</td>
<td>Sets or returns the state of the specified Cascaded S-Parameter stage (enabled or disabled).</td>
</tr>
<tr>
<td>RFGSN:SPAR:CASC:STAG[m]:FILE</td>
<td>Sets or returns the filepath for the specified S-Parameters Cascading Stage, in Cascading mode.</td>
</tr>
<tr>
<td>Command</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td>RFGSignal:SPARameter:CASCading:STAGe[m]:RX[n]</td>
<td>Sets or returns the S-Parameter port assignment of the specified Stage and the channel's specified receiver port number (Rx-Port) in Cascading mode and Single-Ended Signalling Scheme (where applicable).</td>
</tr>
<tr>
<td>RFGSignal:SPARameter:CASCading:STAGe[m]:SSCHEME</td>
<td>Sets or returns the S-Parameter Signalling Scheme, in Cascading mode. Signalling Scheme is only available when the Number of Ports is set to 4, 8, or 12.</td>
</tr>
<tr>
<td>RFGSignal:SPARameter:CASCading:STAGe[m]:TX[n]</td>
<td>Sets or returns the S-Parameter port assignment of the specified Stage and the channel's specified transmission port number (Tx-Port) in Cascading mode and Single-Ended Signalling Scheme (where applicable).</td>
</tr>
<tr>
<td>RFGSignal:SPARameter:CASCading:STYPe</td>
<td>Sets or returns S-Parameter signal type (victim or aggressor), in Cascading mode. The number of ports must be either 8 or 12.</td>
</tr>
<tr>
<td>RFGSignal:SPARameter:CASCading:TYPE</td>
<td>Sets or returns the S-Parameter number of ports, in Cascading mode.</td>
</tr>
<tr>
<td>RFGSignal:SPARameter:MODE</td>
<td>Sets or returns the S-Parameter mode (Cascading or Non-Cascading).</td>
</tr>
<tr>
<td>RFGSignal:SPARameter:NCASCading:AGGRessor2[:ENABLE]</td>
<td>Sets or returns the aggressor 2 signal type state (enabled or disabled) in Non-Cascading mode. Aggressor2 signals are available when the number of ports is set to 12.</td>
</tr>
<tr>
<td>RFGSignal:SPARameter:NCASCading:AGGRessor[n]:AMPLITUDE</td>
<td>Sets or returns the specified Aggressor's amplitude, in Non-Cascading mode.</td>
</tr>
<tr>
<td>RFGSignal:SPARameter:NCASCading:AGGRessor[n]:CTALK</td>
<td>Sets or returns the specified Aggressor's crosstalk type, in Non-Cascading mode.</td>
</tr>
<tr>
<td>RFGSignal:SPARameter:NCASCading:AGGRessor[n]:DRA Te</td>
<td>Sets or returns the specified Aggressor's data rate, in Non-Cascading mode.</td>
</tr>
<tr>
<td>RFGSignal:SPARameter:NCASCading:AGGRessor[n]:SIGNa l</td>
<td>Sets or returns specified Aggressor's signal type, in Non-Cascading mode.</td>
</tr>
<tr>
<td>RFGSignal:SPARameter:NCASCading:AGGRessor[n]:SIGNa l:FILE</td>
<td>Sets or returns the filepath to the aggressor file for the specified Aggressor, in Non-Cascading mode.</td>
</tr>
<tr>
<td>RFGSignal:SPARameter:NCASCading:AGGRessor[n]:SIGNa l:PRBS</td>
<td>Sets or returns the specified Aggressor's PRBS signal type, in Non-Cascading mode.</td>
</tr>
<tr>
<td>RFGSignal:SPARameter:NCASCading:DEEMbed</td>
<td>Sets or returns whether the Non-Cascading S-Parameters is to de-embed (invert) the S-Parameters, in Non-Cascading mode.</td>
</tr>
<tr>
<td>RFGSignal:SPARameter:NCASCading:DRX[n]</td>
<td>Sets or returns the S-Parameter port assignment of the channel's specified receiver port number (Rx-Port) in Non-Cascading mode and Differential Signalling Scheme (where applicable).</td>
</tr>
<tr>
<td>RFGSignal:SPARameter:NCASCading:DTX[n]</td>
<td>Sets or returns the S-Parameter port assignment of the channel's specified transmission port number (Tx-Port) in Non-Cascading mode and Differential Signalling Scheme (where applicable).</td>
</tr>
<tr>
<td>RFGSignal:SPARameter:NCASCading:FILE</td>
<td>Sets or returns the filepath and file name of the S-Parameter file, in Non-Cascading mode.</td>
</tr>
</tbody>
</table>
### Table 2-15: S-Parameter commands and their descriptions (cont.)

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFGSignal:SPARameter:NCAscading:LAYout</td>
<td>Sets or returns the 4 port S-Parameter Matrix Configuration, in Non-Cascading mode.</td>
</tr>
<tr>
<td>RFGSignal:SPARameter:NCAscading:RX[n]</td>
<td>Sets or returns the S-Parameter port assignment of the channel’s specified receiver port number (Rx-Port) in Non-Cascading mode and Single-Ended Signalling Scheme (where applicable).</td>
</tr>
<tr>
<td>RFGSignal:SPARameter:NCAscading:SSCHEME</td>
<td>Sets or returns the S-Parameter Signalling Scheme, in Non-Cascading mode. Signalling Scheme is only available when the Number of Ports is set to 4, 8, or 12.</td>
</tr>
<tr>
<td>RFGSignal:SPARameter:NCAscading:STYPE</td>
<td>Sets or returns S-Parameter signal type (victim or aggressor), in Non-Cascading mode. The number of ports must be either 8 or 12.</td>
</tr>
<tr>
<td>RFGSignal:SPARameter:NCAscading:TX[n]</td>
<td>Sets or returns the S-Parameter port assignment of the channel’s specified transmission port number (Tx-Port) in Non-Cascading mode and Single-Ended Signalling Scheme (where applicable).</td>
</tr>
<tr>
<td>RFGSignal:SPARameter:NCAscading:TYPE</td>
<td>Sets or returns the S-Parameter number of ports, in Non-Cascading mode.</td>
</tr>
</tbody>
</table>

### Sub Carrier Modulation commands

### Table 2-16: Sub Carrier Modulation commands and their descriptions

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFGSignal:SCModulation:AMModulation:AMINDEX</td>
<td>Sets or returns the AM Index for the AM Analog Modulation for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:SCModulation:CFREQUENCY</td>
<td>Sets or returns the Sub Carrier Modulation Carrier Frequency for the carriers in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:SCModulation:FMMODULATION:FDEVIATION</td>
<td>Sets or returns the Frequency Deviation for the FM Analog Modulation for the specified carrier in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:SCModulation:PMMODULATION:PDEVIATION</td>
<td>Sets or returns the Phase Deviation for the PM Sub Carrier Modulation for the carriers in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:SCModulation:TON</td>
<td>Sets or returns the Sub Carrier Modulation state (enabled or disabled) for all carriers in the carrier table.</td>
</tr>
<tr>
<td>RFGSignal:SCModulation:TYPE</td>
<td>Sets or returns the Sub Carrier Modulation type for the carriers in the carrier table.</td>
</tr>
</tbody>
</table>
Commands in alphabetical order

This section contains all available commands. They are presented in alphabetical order.

Use the Command Groups section to simplify navigating to specific groups of commands.

RFGSignal:CARRier:ADD (No Query Form)

This command creates the specified number of carriers in the carrier table. The carriers are created using the default settings and are added to the end of the existing list of carriers.

Group  Carrier setup

Syntax  RFGSignal:CARRier:ADD <carriers>

Arguments  <carriers>::= <NR1> value.

Examples  RFGSIGNAL:CARRIER:ADD 2 adds two carriers to the carrier table.
**RFGSignal:CARRier:COUNt? (Query Only)**

This query returns the carrier count in the carrier table.

**Group**  
Carrier setup

**Syntax**  
RFGSignal:CARRier:COUNt?  
RFGSignal:CARRier:COUNt?

**Returns**  
A single <NR1> value.

**Examples**  
RFGSIGNAL:CARRIER:COUNT? might return 3, indicating that there are 3 carriers in the carrier table.

**RFGSignal:CARRier[n]:AMODulation:AMModulation:AMINdex**

This command sets or returns the AM Index for the AM Analog Modulation for the specified carrier in the carrier table.

**Group**  
Analog modulation

**Syntax**  
RFGSignal:CARRier[n]:AMODulation:AMModulation:AMINdex  
<AM_index>  
RFGSignal:CARRier[n]:AMODulation:AMModulation:AMINdex?

**Arguments**  
[n]::= <NR1> (“n” determines the carrier index number in the carrier table).  
If omitted, n is interpreted as 1.  
<AM_index>::= <NR2> value.

**Returns**  
A single <NR2> value.

**Examples**  
RFGSIGNAL:CARRIER1:AMODULATION:AMMODULATION:AMINDEX 3.5 sets the AM Index as 3.5% for Analog Modulation for the carrier at index 1 in the carrier table.  
RFGSIGNAL:CARRIER3:AMODULATION:AMMODULATION:AMINDEX? might return 30.0000000000, indicating that the AM Index is set to 30° for the carrier at index 3.
RFGSignal:Carrier[n]:AMODulation:FMModulation:FDEVIation

This command sets or returns the Frequency Deviation for the FM Analog Modulation for the specified carrier in the carrier table.

Group          Analog modulation

Syntax         RFGSignal:Carrier[n]:AMODulation:FMModulation:FDEVIation
                <deviation>
                RFGSignal:Carrier[n]:AMODulation:FMModulation:FDEVIation?

Arguments       [n]::= <NR1> ("n" determines the carrier index number in the carrier table).
                If omitted, n is interpreted as 1.
                <deviation>::= <NR3> value.

Returns         A single <NR3> value.

Examples        RFGSIGNAL:CARRIER1:AMODULATION:FMMODULATION:FDEVIATION 1E3
                sets the FM Frequency Deviation to 1 kHz for Analog Modulation for the carrier
                at index 1 in the carrier table.

                return 1.4000000000E+3, indicating that the FM Frequency Deviation is set
                to 1.4 kHz for the carrier at index 3.
**RFGSignal:CARRier[n]:AMODulation:MFRequency**

This command sets or returns the Modulating Frequency of the sinusoidal, triangular, or square Modulating Signals for the specified carrier in the carrier table.

**Conditions**
The Modulating Signal must be set to Sinusoidal, Triangular, or Square before sending this command.

**Group**
Analog modulation

**Syntax**
```
RFGSignal:CARRier[n]:AMODulation:MFRequency <frequency>
RFGSignal:CARRier[n]:AMODulation:MFRequency?
```

**Related Commands**
```
RFGSignal:CARRier[n]:AMODulation:MSIGnal:TYPe
```

**Arguments**

[n]:= <NR1> (“n” determines the carrier index number in the carrier table).
If omitted, n is interpreted as 1.

<frequency>:= <NR3> value.

**Returns**
A single <NR3> value.

**Examples**
```
RFGSIGNAL:CARRIER1:AMODULATION:MFREQUENCY 10E6 sets the Modulating Frequency to 10 MHz for the carrier at index 1 in the carrier table.
RFGSIGNAL:CARRIER3:AMODULATION:MFREQUENCY? might return 1.0000000000E+6, indicating that the Modulating Frequency is set to 1 MHz for the carrier at index 3.
```
RFGSignal:CARRier[n]:AMODulation:MSIGNAL:FNAMe

This command sets or returns the file name for User Defined Modulating Signal for the specified carrier in the carrier table.

Group  Analog modulation

Syntax  RFGSignal:CARRier[n]:AMODulation:MSIGNAL:FNAMe <filepath>
        RFGSignal:CARRier[n]:AMODulation:MSIGNAL:FNAMe?

Arguments  \[n\]::= <NR1> ("n" determines the carrier index number in the carrier table).
            If omitted, n is interpreted as 1.
            <filepath>::= <string> defines path to the modulating signal file.

Returns  A single <filepath> string.

Examples  RFGSIGNAL:CARRIER1:AMODULATION:MSIGNAL:FNAME
          "C:\SAMPLEMODULATINGSCHEME.TXT" sets the modulating signal file’s path and filename for the carrier at index 1 in the carrier table.

          "C:\SampleModulatingScheme.txt" if that file is set to be used for the User-defined analog modulation type for the carrier at index 3.
RFGSignal:CARRier[n]:AMODulation:MSIGnal:INTerpolation

This command sets or returns the Interpolation type for User Defined Modulating Signal for the specified carrier in the carrier table.

Group       Analog modulation

Syntax
RFGSignal:CARRier[n]:AMODulation:MSIGnal:INTerpolation
{SINC|NNEighbor}
RFGSignal:CARRier[n]:AMODulation:MSIGnal:INTerpolation?

Arguments
[n]:= <NR1> (”n” determines the carrier index number in the carrier table).
If omitted, n is interpreted as 1.
SINC
NNEighbor = Nearest Neighbor

Returns
SINC
NNE

Examples
RFGSIGNAL:CARRIER1:AMODULATION:MSIGNAL:INTERPOLATION SINC sets
the Interpolation type to Sinc for the carrier at index 1 in the carrier table.
RFGSIGNAL:CARRIER2:AMODULATION:MSIGNAL:INTERPOLATION? might
return NNE, indicating that the Interpolation type is set to Nearest Neighbor for the
carrier at index 1 in the carrier table.
RFGSignal:CARRier[n]:AMODulation:MSIGNAL:POFFset

This command sets or returns the Phase Offset of the sinusoidal, triangular, or square Modulating Signals for the specified carrier in the carrier table.

**Group**  
Analog modulation

**Syntax**  
RFGSignal:CARRier[n]:AMODulation:MSIGNAL:POFFset
<phase_offset>
RFGSignal:CARRier[n]:AMODulation:MSIGNAL:POFFset?

**Arguments**  
[n]::= <NR1> (“n” determines the carrier index number in the carrier table).  
If omitted, n is interpreted as 1.
<phase_offset>::= <NR3> value.

**Returns**  
A single <NR3> value.

**Examples**  
RFGSIGNAL:CARRIER1:AMODULATION:MSIGNAL:POFFSET 10 sets the modulating signal's phase offset to 10 degrees for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:AMODULATION:MSIGNAL:POFFSET? might return 11.0000000000, indicating that the modulating signal’s phase offset is set to 11 degrees for the carrier at index 3.
RFGSignal:CARRier[n]:AMODulation:MSIGnal:SRATE

This command sets or returns the Sampling Rate for User Defined Modulating Signal for the specified carrier in the carrier table.

**Group**  Analog modulation

**Syntax**

RFGSignal:CARRier[n]:AMODulation:MSIGnal:SRATE
<sampling_rate>
RFGSignal:CARRier[n]:AMODulation:MSIGnal:SRATE?

**Arguments**

[n]:= <NR1> (“n” determines the carrier index number in the carrier table).
If omitted, n is interpreted as 1.

<sampling_rate>: = <NR3> value.

**Returns**

A single <NR3> value.

**Examples**

RFGSIGNAL:CARRIER1:AMODULATION:MSIGNAL:SRATE 100E6 sets the User Defined modulating signal's sampling rate to 100 MS/s for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:AMODULATION:MSIGNAL:SRATE? might return 150.0000000000E+6, indicating that the user defined modulating signal's sampling rate is set to 150 MHz for the carrier at index 3.
RFGSignal: CARRIER[n]: AMODulation: MSIGNAL: TYPE

This command sets or returns the Modulating Signal type for Analog Modulation for the specified carrier in the carrier table.

Group  Analog modulation

Syntax  RFGSignal: CARRIER[n]: AMODulation: MSIGNAL: TYPE
        {SINusoidal|TRIangular|SQUare|UDEFined}
        RFGSignal: CARRIER[n]: AMODulation: MSIGNAL: TYPE?

Arguments  [n]:= <NR1> ("n" determines the carrier index number in the carrier table).
            If omitted, n is interpreted as 1.

            SINusoidal = Sinusoidal
            TRIangular = Triangular
            SQUare = Square
            UDEFined = User Defined

Returns  SIN
         TRI
         SQU
         UDEF

Examples  RFGSIGNAL: CARRIER1: AMODULATION: MSIGNAL: TYPE SIN sets the Modulating Signal type to SINUSOIDAL for the carrier at index 1 in the carrier table.

            RFGSIGNAL: CARRIER3: AMODULATION: MSIGNAL: TYPE? might return UDEF, indicating that the Modulating Signal type is set to User Defined for the carrier at index 3.
RFGSignal:CARRier[n]:AMODulation:PMModulation:PDEViation

This command sets or returns the Phase Deviation for the PM Analog Modulation for the specified carrier in the carrier table.

**Group**
Analog modulation

**Syntax**
RFGSignal:CARRier[n]:AMODulation:PMModulation:PDEViation

<phase_deviation>

RFGSignal:CARRier[n]:AMODulation:PMModulation:PDEViation?

**Arguments**

[n]:= <NR1> (“n” determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

<phase_deviation>::= <NR3> value.

**Returns**
A single <NR2> value.

**Examples**

RFGSIGNAL:CARRIER1:AMODULATION:PMMODULATION:PDEVIATION 11.5
sets the PM Phase Deviation to 11.5° for Analog Modulation for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER2:AMODULATION:PMMODULATION:PDEVIATION? might return 10.0000000000, indicating that the PM Phase Deviation is set to 10° for the carrier at index 3.
RFGSignal:CARRier[n]:AMODulation:TYPe

This command sets or returns the Analog Modulation type for the specified carrier in the carrier table.

Group          Analog modulation

Syntax          RFGSignal:CARRier[n]:AMODulation:TYPe {AM|FM|PM}
                RFGSignal:CARRier[n]:AMODulation:TYPe?

Related Commands RFGSignal:CARRier[n]:TYPE

Arguments        [n]::= <NR1> (“n” determines the carrier index number in the carrier table).
                  If omitted, n is interpreted as 1.

AM, FM, and PM are the Analog Modulation types.

Returns          AM, FM, or PM indicating the analog modulation type.

Examples         RFGSIGNAL:CARRIER1:AMODULATION:TYPe FM sets the Analog Modulation type to FM for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:AMODULATION:TYPe? might return PM, indicating that the Analog Modulation type is set to Phase Modulation for the carrier at index 3.
**RFGSignal:Carrier[n]:BBOFfset**

This command sets or returns the base band Offset for the specified carrier in the carrier table.

**Conditions**
Applicable when the signal type is set to IQ.

**Group**
Carrier setup

**Syntax**

```
RFGSignal:Carrier[n]:BBOFfset <offset>
RFGSignal:Carrier[n]:BBOFfset?
```

**Arguments**

\[n\]::= \text{<NR1>} ("n" determines the carrier index number in the carrier table).

If omitted, \(n\) is interpreted as 1.

\(<\text{offset}>\)::= \text{<NR3>} value.

**Returns**
A single \text{<NR3>} value.

**Examples**

```
RFGSIGNAL:CARRIER1:BBOFFSET 2E9 sets the base band Offset to 2 GHz for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:BBOFFSET? might return 4.0000000000E+9, indicating that the base band Offset frequency is set to 4 GHz for the carrier at index 3.
```
RFGSignal:CARRier[n]:CMODulation:ADDMap (No Query Form)

This command adds the specified number of maps to the Custom Modulation table for the specified carrier in the carrier table.

**Group**  
Custom modulation

**Syntax**  
RFGSignal:CARRier[n]:CMODulation:ADDMap <maps>

**Arguments**  
[n]:= <NR1> (“n” determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

<maps>:= <NR1> value.

**Examples**  
RFGSIGNAL:CARRIER1:CMODULATION:ADDMAP 2 adds 2 new maps at the end of the Custom Modulation table for the carrier at index 1 in the carrier table.
**RFGSignal:CARRier[n]:CMODulation:DELMap (No Query Form)**

This command deletes a single Custom Modulation map for the specified carrier in the carrier table.

**Group**  
Custom modulation

**Syntax**  
RFGSignal:CARRier[n]:CMODulation:DELMap <index>

**Arguments**  
[n]::= <NR1> (“n” determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

<index>::= <NR1> value.

**Examples**  
RFGSIGNAL:CARRIER1:CMODULATION:DELMAP  2 deletes map 2 in the Custom Modulation table for the carrier at index 1 in the carrier table.
RFGSignal:CARRier[n]:CMODulation:ISYMbol

This command sets or returns the I symbol value of the currently selected Custom Modulation map for the specified carrier in the carrier table.

Group Custom modulation

Syntax RFGSignal:CARRier[n]:CMODulation:ISYMbol <I_symbol>

Related Commands RFGSignal:CARRier[n]:CMODulation:SELMap

Arguments [n] ::= <NR1> ("n" determines the carrier index number in the carrier table).
If omitted, n is interpreted as 1.

<I_symbol> ::= <NR2> value.

Returns A single <NR2> value.

Examples RFGSIGNAL:CARRIER1:CMODULATION:ISYMBOL 25 sets the I symbol value of the currently selected map to 25 for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:CMODULATION:ISYMBOL? might return 5.0000000000, indicating that the I symbol value of the currently selected map is set to 5 for the carrier at index 3.
RFGSignal:CARRier[n]:CMODulation:MMODE

This command sets or returns the Modulation mode for the Custom Modulation for the specified carrier in the carrier table.

**Group**
Custom modulation

**Syntax**
RFGSignal:CARRier[n]:CMODulation:MMODE {NORMal|DIFFerential}
RFGSignal:CARRier[n]:CMODulation:MMODE?

**Related Commands**
RFGSignal:CARRier[n]:TYPE

**Arguments**
[n]:= <NR1> (“n” determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

NORMal

DIFFerential

**Returns**
NORM

DIFF

**Examples**
RFGSIGNAL:CARRIER1:CMODULATION:MMODE NORM sets the Modulation mode for Custom Modulation to Normal for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:CMODULATION:MMODE? might return DIFF, indicating that the Modulation mode for Custom Modulation is set to Differential for the carrier at index 3.
RFGSignal:CARRier[n]:CMODulation:MOFFset

This command sets or returns the Offset modulation state (enabled or disabled) for Custom Modulation for the specified carrier in the carrier table.

**Group**
Custom modulation

**Syntax**
RFGSignal:CARRier[n]:CMODulation:MOFFset {1|0|ON|OFF}
RFGSignal:CARRier[n]:CMODulation:MOFFset?

**Arguments**

[n]::= <NR1> ("n" determines the carrier index number in the carrier table).
If omitted, n is interpreted as 1.
OFF or 0 disables Offset modulation. OFF or 0 is the default value.
ON or 1 enables Offset modulation.

**Returns**
A single <Boolean> value.

**Examples**

RFGSIGNAL:CARRIER1:CMODULATION:MOFFSET 1 enables Offset modulation for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:CMODULATION:MOFFSET? might return 0, indicating that Offset modulation is disabled for the carrier at index 3.
Commands in alphabetical order

RFGSignal:CARRier[n]:CMODulation:QSYMbol

This command sets or returns the Q symbol value of the currently selected Custom Modulation map for the specified carrier in the carrier table.

Group Custom modulation

Syntax RFGSignal:CARRier[n]:CMODulation:QSYMbol <Q_symbol>

Related Commands RFGSignal:CARRier[n]:CMODulation:SELMap

Arguments

[n]::= <NR1> (“n” determines the carrier index number in the carrier table).
If omitted, n is interpreted as 1.

<Q_symbol>::= <NR2> value.

Returns A single <NR2> value.

Examples

RFGSIGNAL:CARRIER1:CMODULATION:QSYMBOL 25 sets the Q symbol value of the currently selected map to 25 for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:CMODULATION:QSYMBOL? might return 5.0000000000, indicating that the Q symbol value of the currently selected map is set to 5 for the carrier at index 3.
RFGSignal:CARRier[n]:CMODulation:SELMap

This command sets or returns the selected map in the Custom Modulation table for the specified carrier in the carrier table.

Group  Custom modulation

Syntax  
RFGSignal:CARRier[n]:CMODulation:SELMap <index>
RFGSignal:CARRier[n]:CMODulation:SELMap?

Arguments  
[n]::= <NR1> (“n” determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

<index>::= <NR1> value.

Returns  A single <NR1> value.

Examples  
RFGSIGNAL:CARRIER1:CMODULATION:SELMAP 3 selects the third map in the Custom Modulation table for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:CMODULATION:SELMAP? might return 2.0000000000, indicating that the second map in the Custom Modulation table is selected for the carrier at index 3.
RFGSignal:CARRier[n]:CMODulation:SRATe

This command sets or returns the Symbol Rate for the Custom Modulation for the specified carrier in the carrier table.

**Group**
Custom modulation

**Syntax**

RFGSignal:CARRier[n]:CMODulation:SRATe <rate>
RFGSignal:CARRier[n]:CMODulation:SRATe?

**Related Commands**
RFGSignal:CARRier[n]:TYPE

**Arguments**

[n]::= <NR1> (“n” determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

<rate>::= <NR3> value.

**Returns**
A single <NR3> value.

**Examples**

RFGSIGNAL:CARRIER1:CMODULATION:SRATE 10E6 sets the Custom Modulation Symbol Rate to 10 MHz for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:CMODULATION:SRATE? might return 11.0000000000E+6, indicating that the Custom Modulation Symbol Rate is set to 11 MHz for the carrier at index 3.
RFGSignal:CARRier[n]:DATA

This command sets or returns the Data Pattern type for the specified carrier in the carrier table. When setting the pattern type to PRBS, Pattern, or File, use the additional commands noted with each argument to continue defining these pattern types.

**Group**
Data

**Syntax**
RFGSignal:CARRier[n]:DATA {AZERO|AONE|PRBS|PATTERN|FILE}
RFGSignal:CARRier[n]:DATA?

**Arguments**
[n]::= <NR1> (“n” determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

AZERO is a data type of all zeroes.
AONE is a data type of all ones.

PRBS is one of the available PRBS data types. Use the command RFGSignal:CARRier[n]:DATA:PRBS to set and query the actual PRBS data type.

PATTERN is a pattern data type. Use the command RFGSignal:CARRier[n]:DATA:FILE to set and query the pattern.

FILE is a file data type. Use the command RFGSignal:CARRier[n]:DATA:FILE to set and query the pattern file and file path.

**Returns**
AZER
AON
PRBS
PATT
FIL

**Examples**
RFGSIGNAL:CARRIER1:DATA AONE sets the Data Type to All ones for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:DATA? might return AONE, indicating that the Data Type is set to all ones for the carrier at index 3.
RFGSignal:CARRIER[n]:DATA:CODing

This command sets or returns the coding type for the specified carrier in the carrier table.

Group

Data

Syntax

RFGSignal:CARRIER[n]:DATA:CODing {NONE|GRAY|DIFFerential}
RFGSignal:CARRIER[n]:DATA:CODing?

Arguments

[n]::= <NR1> (“n” determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

NONE: No coding applied.
GRAY: Applies gray coding to the information bits.
DIFFerential: Applies differential coding to the information bits.

Returns

NONE
GRAY
DIFFerential

Examples

RFGSIGNAL:CARRIER1:DATA:CODING GRAY sets the coding to gray for the carrier at index 1 in the carrier table.
RFGSIGNAL:CARRIER3:DATA:CODING? might return NON, indicating no coding is applied to the carrier at index 3.
**RFGSignal:CARRier[n]:DATA:FILE**

This command sets or returns the pattern file and file path for the specified carrier in the carrier table.

**Conditions**
The Data Pattern type must be set to File with the command **RFGSignal:CARRIER[n]:DATA**.

**Group**
Data

**Syntax**
RFGSignal:CARRIER[n]:DATA:FILE <filepath>

**Related Commands**
RFGSignal:CARRier[n]:DATA

**Arguments**

[n]:= <NR1> ("n" determines the carrier index number in the carrier table).
If omitted, n is interpreted as 1.

<filepath>::= <string>.

**Returns**
A single <filepath> string.

**Examples**
RFGSIGNAL:CARRIER1:DATA:FILE "C:\temp\dataPattern.txt" sets the pattern file and path for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:DATA:FILE? might return "C:\temp\dataPattern.txt" as the filepath for the carrier at index 3.
RFGSignal:CARRier[n]:DATA:PATTern

This command sets or returns the user defined data pattern for the specified carrier in the carrier table.

**Conditions**
The Data Pattern type must be set to Pattern with the command RFGSignal:CARRier[n]:DATA.

**Group**
Data

**Syntax**
RFGSignal:CARRier[n]:DATA:PATTern <pattern>

**Related Commands**
RFGSignal:CARRier[n]:DATA

**Arguments**
[n]::= <NR1> (“n” determines the carrier index number in the carrier table).
If omitted, n is interpreted as 1.
<pattern> ::= <string>.

**Returns**
A single <pattern> string.

**Examples**
RFGSIGNAL:CARRIER1:DATA:PATTERN "11001111" sets the data pattern for the carrier at index 1 in the carrier table.
RFGSIGNAL:CARRIER3:DATA:PATTERN? might return "11001101", indicating this is the set data pattern for the carrier at index 3.
RFGSignal:CARRier[n]:DATA:PRBS

This command sets or returns the PRBS data type when the data Pattern is set to PRBS for the specified carrier in the carrier table.

Group
Data

Syntax
RFGSignal:CARRier[n]:DATA:PRBS {PRBS7|PRBS9|PRBS15|PRBS16
|PRBS20|PRBS21|PRBS23|PRBS29|PRBS31|UDEFined}
RFGSignal:CARRier[n]:DATA:PRBS?

Arguments

[n]::= <NR1> (“n” determines the carrier index number in the carrier table).
If omitted, n is interpreted as 1.

PRBS7, PRBS9, PRBS15, PRBS16, PRBS20, PRBS21, PRBS23, PRBS29,
PRBS31, and UDEFined are the PRBS types.

When setting to UDEF (User Defined), use the commands
RFGSignal:CARRier[n]:DATA:PRBS:UDEFined:POLYnomial and
RFGSignal:CARRier[n]:DATA:PRBS:UDEFined:SREGister
to define the PRBS Polynomial Expression and Shift register initial value.

Returns
PRBS7, PRBS9, PRBS15, PRBS16, PRBS20, PRBS21, PRBS23, PRBS29,
PRBS31, UDEF

Examples
RFGSIGNAL:CARRIER1:DATA:PRBS PRBS15 sets the PRBS Data Type to PRBS 15 for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:DATA:PRBS? might return PRBS15, indicating that the
PRBS Data Type is set to PRBS15 for the carrier at index 3.
RFGSignal:CARRier[n]:DATA:PRBS:UDEFined:POLYnomial

This command sets or returns the PRBS Polynomial Expression for the User Defined PRBS for the specified carrier in the carrier table.

**Group**  
Data

**Syntax**  
RFGSignal:CARRier[n]:DATA:PRBS:UDEFined:POLYnomial
<polynomial>
RFGSignal:CARRier[n]:DATA:PRBS:UDEFined:POLYnomial?

**Arguments**  
[n]:= <NR1> ("n" determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

<polynomial>::= <string>.

The highest degree of polynomial is 31. If the expression contains more than 31 an error will be returned. For example X32+X23+1 returns an error.

**Returns**  
A single <polynomial> string.

**Examples**  
RFGSIGNAL:CARRIER1:DATA:PRBS:UDEFINED:POLYNOMIAL "X12+X11+1" sets the User Defined PRBS Polynomial Expression to X12+X11+1 for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:DATA:PRBS:UDEFINED:POLYNOMIAL? might return "X12+X11+1", indicating this is the User Defined PRBS Polynomial Expression for the carrier at index 3.
RFGSignal:CARRier[n]:DATA:PRBS:UDEFined:RESet (No Query Form)

This command resets the User Defined PRBS polynomial expression and shift register to their default values for the specified carrier in the carrier table.

Group     Data

Syntax    RFGSignal:CARRier[n]:DATA:PRBS:UDEFined:RESet

Arguments [n]:= <NR1> (“n” determines the carrier index number in the carrier table).
            If omitted, n is interpreted as 1.

Examples  RFGSIGNAL:CARRIER1:DATA:PRBS:UDEFINED:RESET resets the User Defined PRBS polynomial expression and shift register values for the carrier at index 1 in the carrier table.
RFGSignal:CARRier[n]:DATA:PRBS:UDEFined:SREGister

This command sets or returns the PRBS Shift register initial value for the User Defined PRBS for the specified carrier in the carrier table.

**Group**  
Data

**Syntax**  
RFGSignal:CARRier[n]:DATA:PRBS:UDEFined:SREGister  
<shift_register>  
RFGSignal:CARRier[n]:DATA:PRBS:UDEFined:SREGister?

**Arguments**  
[n]::= <NR1> (“n” determines the carrier index number in the carrier table).  
If omitted, n is interpreted as 1.  
<shift_register>::= <string>.  
The number of shift register values should match the highest polynomial degree.  
For example, if the user defined polynomial is "X5+X4+1", then the initial shift register should have 5 values.

**Returns**  
A single <shift_register> string.

**Examples**  
RFGSIGNAL:CARRIER1:DATA:PRBS:UDEFINED:SREGISTER "11011" sets the User Defined PRBS Shift register to 11011 for the carrier at index 1 in the carrier table.  
RFGSignal:CARRier[n]:DELeTe (No Query Form)

This command deletes the specified carrier from the carrier table.

Group  Carrier setup

Syntax  RFGSignal:CARRier[n]:DELeTe

Related Commands  RFGSignal:CARRier:COUNt?

Arguments  [n]::= <NR1> (“n” determines the carrier index number in the carrier table).
            If omitted, n is interpreted as 1.

Examples  RFGSignal:CARRier3:DELeTe deletes carrier 3 from the carrier table.
**RFGSignal:CARRier[n]:DISTortion:AMPLifier:AM3K**

This command sets or returns the k3 level for AM/AM Distortions for the specified carrier in the carrier table.

The Distortion type must be set to Non linear.

**Group**
Distortion

**Syntax**

RFGSignal:CARRier[n]:DISTortion:AMPLifier:AM3K <level>
RFGSignal:CARRier[n]:DISTortion:AMPLifier:AM3K?

**Related Commands**
RFGSignal:CARRier[n]:DISTortion:AMPLifier:TYPE

**Arguments**

[n]::= <NR1> (“n” determines the carrier index number in the carrier table).
If omitted, n is interpreted as 1.

[level]::= <NR3> value.

**Returns**
A single <NR3> value.

**Examples**

RFGSIGNAL:CARRIER1:DISTORTION:AMPLIFIER:AM3K -1 sets the AM/AM k3 level to –1 dB for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:DISTORTION:AMPLIFIER:AM3K? might return -1.0000000000, indicating that the AM/AM k3 level is set to –1 dB for the carrier at index 3.
RFGSignal:CARRier[n]:DISTortion:AMPLifier:AM5K

This command sets or returns the k5 level for AM/AM Distortions for the specified carrier in the carrier table.

The Distortion type must be set to Non linear.

Group     Distortion

Syntax     RFGSignal:CARRier[n]:DISTortion:AMPLifier:AM5K <level>
            RFGSignal:CARRier[n]:DISTortion:AMPLifier:AM5K?

Related Commands  RFGSignal:CARRier[n]:DISTortion:AMPLifier:TYPE

Arguments [n]:= <NR1> (“n” determines the carrier index number in the carrier table).
            If omitted, n is interpreted as 1.
            <level>:= <NR3> value.

Returns    A single <NR3> value.

Examples  RFGSIGNAL:CARRIER1:DISTORTION:AMPLIFIER:AM5K -1 sets the AM/AM k5 level to –1 dB for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:DISTORTION:AMPLIFIER:AM5K? might return –1.0000000000, indicating that the AM/AM k5 level is set to –1 dB for the carrier at index 3.
RFGSignal:CARRier[n]:DISTortion:AMPLifier:LLEVEL

This command sets or returns the Limiting Level value of amplifier distortion for the specified carrier in the carrier table.

The Distortion type must be set to Soft Limiting.

**Group**  
Distortion

**Syntax**  
RFGSignal:CARRier[n]:DISTortion:AMPLifier:LLEVEL <level>  
RFGSignal:CARRier[n]:DISTortion:AMPLifier:LLEVEL?

**Related Commands**  
RFGSignal:CARRier[n]:DISTortion:AMPLifier:TYPe

**Arguments**  
[n]:= <NR1> (“n” determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

<level>::= <NR3> value.

**Returns**  
A single <NR3> value.

**Examples**  
RFGSIGNAL:CARRIER1:DISTORTION:AMPLIFIER:LLEVEL 10 sets the Limiting Level value to 10 for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER1:DISTORTION:AMPLIFIER:LLEVEL? might return 100.0000000000E-3, indicating that the Limiting Level value is set to 0.1 for the carrier at index 3.
RFGSignal:CARRier[n]:DISTortion:AMPLifier:OPOInt

This command sets or returns the Operating Point value of amplifier distortion for the specified carrier in the carrier table.

The Distortion type must be set to Non linear.

Group       Distortion
Syntax      RFGSignal:CARRier[n]:DISTortion:AMPLifier:OPOInt <value>
            RFGSignal:CARRier[n]:DISTortion:AMPLifier:OPOInt?
Related Commands  RFGSignal:CARRier[n]:DISTortion:AMPLifier:TYPE
Arguments      [n]::= <NR1> (“n” determines the carrier index number in the carrier table).
                If omitted, n is interpreted as 1.
                <value>::= <NR3> value.
Returns        A single <NR3> value.
Examples       RFGSIGNAL:CARRIER1:DISTORTION:AMPLIFIER:OPOINT 10 sets the Operating Point value to 10 for the carrier at index 1 in the carrier table.
                RFGSIGNAL:CARRIER3:DISTORTION:AMPLIFIER:OPOINT? might return 15, indicating that the Operating Point value is set to 3 % for the carrier at index 3.
**RFGSignal:CARRier[n]:DISTortion:AMPLifier:PM3K**

This command sets or returns the k3 level for AM/PM Distortions for the specified carrier in the carrier table.

The Distortion type must be set to Non linear.

**Group**
Distortion

**Syntax**

```plaintext
RFGSignal:CARRier[n]:DISTortion:AMPLifier:PM3K <level>
RFGSignal:CARRier[n]:DISTortion:AMPLifier:PM3K?
```

**Related Commands**

`RFGSignal:CARRier[n]:DISTortion:AMPLifier:TYPE`  

**Arguments**

[n]::= <NR1> ("n" determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

<level>::= <NR3> value.

**Returns**

A single <NR> value.

**Examples**

`RFGSIGNAL:CARRIER1:DISTORTION:AMPLIFIER:PM3K -1` sets the AM/PM k3 level to –1 dB for the carrier at index 1 in the carrier table.

`RFGSIGNAL:CARRIER3:DISTORTION:AMPLIFIER:PM3K?` might return -1.0000000000, indicating that the AM/PM k3 level is set to –1 dB for the carrier at index 3.
**RFGSignal:CARRier[n]:DISTortion:AMPLifier:PM5K**

This command sets or returns the k5 level for AM/PM Distortions for the specified carrier in the carrier table.

The Distortion type must be set to Non linear.

**Group**

Distortion

**Syntax**

RFGSignal:CARRier[n]:DISTortion:AMPLifier:PM5K <level>

RFGSignal:CARRier[n]:DISTortion:AMPLifier:PM5K?

**Related Commands**

RFGSignal:CARRier[n]:DISTortion:AMPLifier:TYPE

**Arguments**

[n]: = <NR1> (“n” determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

[level]: = <NR3> value.

**Returns**

A single <NR3> value.

**Examples**

RFGSIGNAL:CARRIER1:DISTORTION:AMPLIFIER:PM5K -1 sets the AM/PM k5 level to –1 dB for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:DISTORTION:AMPLIFIER:PM5K? might return -1.0000000000, indicating that the AM/PM k5 level is set to –1 dB for the carrier at index 3.
RFGSignal:CARRier[n]:DISTortion:AMPLifier:TON

This command sets or returns the Distortion state (enabled or disabled) for the specified carrier in the carrier table.

**Group**  
Distortion

**Syntax**  
RFGSignal:CARRier[n]:DISTortion:AMPLifier:TON {1|0|ON|OFF}  
RFGSignal:CARRier[n]:DISTortion:AMPLifier:TON?

**Arguments**  
[n]:= <NR1> ("n" determines the carrier index number in the carrier table).  
If omitted, n is interpreted as 1.  
OFF or 0 disables Distortion. OFF or 0 is the default value.  
ON or 1 enables Distortion.

**Returns**  
A single <Boolean> value.

**Examples**  
RFGSIGNAL:CARRIER1:DISTORTION:AMPLIFIER:TON 1 enables the Distortion for the carrier at index 1 in the carrier table.  
RFGSIGNAL:CARRIER3:DISTORTION:AMPLIFIER:TON? might return 0, indicating that the Distortion is disabled for the carrier at index 3.
RFGSignal:CARRIER[n]:DISTortion:AMPLifier:TYPE

This command sets or returns the amplifier Distortion type for the specified carrier in the carrier table.

**Group**  Distortion

**Syntax**

RFGSignal:CARRIER[n]:DISTortion:AMPLifier:TYPE
{NLINear|SLIMiting|HLIMiting}
RFGSignal:CARRIER[n]:DISTortion:AMPLifier:TYPE

**Arguments**

[n]::= \(<\text{NR1}\>\) ("n" determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

NLINear = Non linear.

SLIMiting = Soft Limiting.

HLIMiting = Hard Limiting.

**Returns**

NLIN

SLIM

HLIM

**Examples**

RFGSIGNAL:CARRIER1:DISTORTION:AMPLIFIER:TYPE NLINear sets the amplifier Distortion type to Non Linear for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER1:DISTORTION:AMPLIFIER:TYPE? might return SLIM, indicating that the amplifier Distortion type is set to Soft Limiting for the carrier at index 3.
RFGSignal:CARRier[n]:DMODulation:APSK

This command sets or returns the APSK Modulation type for the specified carrier in the carrier table.

APSK Modulation requires further definition. See related commands.

Group  Digital modulation

Syntax  RFGSignal:CARRier[n]:DMODulation:APSK {APSK16|APSK32|APSK64}
        RFGSignal:CARRier[n]:DMODulation:APSK?

Related Commands  RFGSignal:CARRier[n]:DMODulation:APSK:SRINg
                    RFGSignal:CARRier[n]:DMODulation:APSK:SYMBols
                    RFGSignal:CARRier[n]:DMODulation:APSK:RADius
                    RFGSignal:CARRier[n]:DMODulation:APSK:PROTation

Arguments  [n]:= <NR1> (“n” determines the carrier index number in the carrier table).
           If omitted, n is interpreted as 1.
           APSK16, APSK32, and APSK64 are the selectable APSK digital modulation types.

Returns  APSK16, APSK32, or APSK64 indicating the APSK digital modulation type.

Examples  RFGSIGNAL:CARRIER1:DMODULATION:APSK APSK32 sets the APSK Modulation type to 32 APSK for the carrier at index 1 in the carrier table.
          RFGSIGNAL:CARRIER3:DMODULATION:APSK? might return APSK32, indicating that the APSK Modulation type is set to 32 APSK for the carrier at index 3.
RFGSignal:CARRier[n]:DMODulation:APSK:PROTation

This command sets or returns the Phase Rotation of the selected APSK Ring Index for the specified carrier in the carrier table.

Use the command RFGSignal:CARRier[n]:DMODulation:APSK:SRINg to set or query the selected APSK Ring Index setting.

Group Digital modulation

Syntax

RFGSignal:CARRier[n]:DMODulation:APSK:PROTation <rotation>
RFGSignal:CARRier[n]:DMODulation:APSK:PROTation?

Related Commands

RFGSignal:CARRier[n]:DMODulation:APSK:SRINg

Arguments

[n]:= <NR1> ("n" determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

<rotation>::= <NR1> value.

Returns

A single <NR1> value.

Examples

RFGSIGNAL:CARRIER1:DMODULATION:APSK:PROTATION 45 sets the selected APSK Phase Rotation to 45 degrees for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:DMODULATION:APSK:PROTATION? might return 180.0000000000, indicating that the selected APSK Phase Rotation is set to 180 degrees for the carrier at index 3.
RFGSignal:CARRier[n]:DMODulation:APSK:RADius

This command sets or returns the Radius of the selected APSK Ring Index for the specified carrier in the carrier table.

Use the command RFGSignal:CARRier[n]:DMODulation:APSK:SRINg to set or query the selected APSK Ring Index setting.

**Group**
Digital modulation

**Syntax**
RFGSignal:CARRier[n]:DMODulation:APSK:RADius <radius>
RFGSignal:CARRier[n]:DMODulation:APSK:RADius?

**Related Commands**
RFGSignal:CARRier[n]:DMODulation:APSK:SRINg

**Arguments**
[n]:= <NR1> (“n” determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

<radius>:= <NR2> value.

**Returns**
A single <NR2> value.

**Examples**
RFGSIGNAL:CARRIER1:DMODULATION:APSK:RADIUS 10 sets the selected APSK Radius to 10 for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:DMODULATION:APSK:RADIUS? might return 4.5800000000, indicating that the selected APSK Radius is set to 4.58 for the carrier at index 3.
RFGSignal:CARRier[n]:DMODulation:APSK:SRING

This command sets or returns the selected APSK Ring Index for the specified carrier in the carrier table.

Group  
Digital modulation

Syntax  
RFGSignal:CARRier[n]:DMODulation:APSK:SRING <index>
RFGSignal:CARRier[n]:DMODulation:APSK:SRING?

Arguments  
[n]::= <NR1> (“n” determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

:index>::= <NR1> value.

Returns  
A single <NR1> value.

Examples  
RFGSIGNAL:CARRIER1:DMODULATION:APSK:SRING 2 sets the selected APSK Ring Index to 2 for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:DMODULATION:APSK:SRING? might return 2.0000000000, indicating that the selected APSK Ring Index is 2 for the carrier at index 3.
**RFGSignal:CARRIER[n]:DMODulation:APSK:SYMBols**

This command sets or returns the Number of Symbols of the selected APSK Ring Index for the specified carrier in the carrier table.

Use the command `RFGSignal:CARRIER[n]:DMODulation:APSK:SRING` to set or query the selected APSK Ring Index setting.

**Group**
Digital modulation

**Syntax**
```
RFGSignal:CARRIER[n]:DMODulation:APSK:SYMBols <symbols>
RFGSignal:CARRIER[n]:DMODulation:APSK:SYMBols?
```

**Arguments**

- `[n]`::= `<NR1>` ("n" determines the carrier index number in the carrier table).
  
  If omitted, `n` is interpreted as 1.

- `<symbols>`::= `<NR1>` value.

**Returns**
A single `<NR1>` value.

**Examples**

- `RFGSIGNAL:CARRIER1:DMODULATION:APSK:SYMBOLS 12` sets the APSK Number of Symbols to 12 for the selected APSK ring index.

- `RFGSIGNAL:CARRIER3:DMODULATION:APSK:SYMBOLS?` might return 12.0000000000, indicating that the selected APSK Symbol count is set to 12 for the carrier at index 3.
RFGSignal:CARRier[n]:DMODulation:ASK:MINDex

This command sets or returns the ASK Mod Index for ASK modulation for the specified carrier in the carrier table.

**Group**
Digital modulation

**Syntax**
RFGSignal:CARRier[n]:DMODulation:ASK:MINDex <mod_index>
RFGSignal:CARRier[n]:DMODulation:ASK:MINDex?

**Related Commands**
RFGSignal:CARRier[n]:DMODulation:TYPe

**Arguments**
[n]::= <NR1> ("n" determines the carrier index number in the carrier table).
If omitted, n is interpreted as 1.

<mod_index> ::= <NR2> value.

**Returns**
A single <NR2> value.

**Examples**
RFGSIGNAL:CARRIER1:DMODULATION:ASK:MINDEX 10 sets the ASK Mod Index value to 10% for the carrier at index 1 in the carrier table.
RFGSIGNAL:CARRIER3:DMODULATION:ASK:MINDEX? might return 2.5000000000, indicating that the ASK Mod Index value is set to 2.5% for the carrier at index 3.
RFGSignal:CARRier[n]:DMODulation:CPM:MINDex

This command sets or returns the Index for CPM modulation for the specified carrier in the carrier table.

Group
Digital modulation

Syntax
RFGSignal:CARRier[n]:DMODulation:CPM:MINDex {1|2|3|4|5|6}
RFGSignal:CARRier[n]:DMODulation:CPM:MINDex?

Arguments
[n]:= <NR1> (“n” determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

1 = 4/16, 5/16
2 = 5/16, 6/16
3 = 6/16, 7/16
4 = 7/16, 10/16
5 = 12/16, 13/16
6 = 8/16, 8/16

Returns
1, 2, 3, 4, 5, or 6 indicating the CPM modulation index.

Examples
RFGSIGNAL:CARRIER1:DMODULATION:CPM:MINDEX 2 sets the CPM modulation type to 5/16, 6/16 for the carrier at index 1 in the carrier table.
RFGSIGNAL:CARRIER3:DMODULATION:CPM:MINDEX? might return 4.0000000000, indicating that the CPM modulation index value is set to 7/16, 10/16 for the carrier at index 3.
RFGSignal:CARRier[n]:DMODulation:FSK

This command sets or returns the FSK Modulation type for the specified carrier in the carrier table.

Group Digital modulation

Syntax
RFGSignal:CARRier[n]:DMODulation:FSK
{FSK2|FSK4|FSK8|FSK16|FSK32}
RFGSignal:CARRier[n]:DMODulation:FSK?

Arguments
[n]:= <NR1> ("n" determines the carrier index number in the carrier table).
If omitted, n is interpreted as 1.
FSK2, FSK4, FSK8, FSK16, and FSK32 are the FSK modulation types.

Returns
FSK2, FSK4, FSK8, FSK16, or FSK32 indicating the FSK modulation type.

Examples
RFGSIGNAL:CARRIER1:DMODULATION:FSK FSK32 sets the FSK modulation type to 32 FSK for the carrier at index 1 in the carrier table.
RFGSIGNAL:CARRIER3:DMODULATION:FSK? might return FSK32, indicating that the FSK Modulation type is set to 32 FSK for the carrier at index 3.
RFGSignal:CARRier[n]:DMODulation:FSK:PDEViation

This command sets or returns the FSK Peak Deviation for the specified carrier in the carrier table.

Group  Digital modulation

Syntax  RFGSignal:CARRier[n]:DMODulation:FSK:PDEViation
        <peak_deviation>
        RFGSignal:CARRier[n]:DMODulation:FSK:PDEViation?

Arguments  [n]::= <NR1> (“n” determines the carrier index number in the carrier table).
           If omitted, n is interpreted as 1.
           <peak_deviation>::= <NR3> value.

Returns  A single <NR3> value.

Examples  RFGSIGNAL:CARRIER1:DMODULATION:FSKPDEVIATION 1E6 sets the FSK Peak Deviation to 1 MHz for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:DMODULATION:FSKPDEVIATION? might return 2.0000000000E+6, indicating that the FSK Peak Deviation is set to 2 MHz for the carrier at index 3.
RFGSignal:CARRier[n]:DMODulation:NDPSK:INDEX

This command sets or returns the ‘n’ value (the kind of DPSK) for nDPSK modulation for the specified carrier in the carrier table.

Group: Digital modulation

Syntax:

RFGSignal:CARRier[n]:DMODulation:NDPSK:INDEX <n_value>
RFGSignal:CARRier[n]:DMODulation:NDPSK:INDEX?

Related Commands:
RFGSignal:CARRier[n]:DMODulation:TYPe

Arguments:

[n]:= <NR1> (“n” determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

<n_value>:= <NR1> value from 2 to 128 in multiples of 2.

Returns:
A single <NR2> value.

Examples:
RFGSIGNAL:CARRIER1:DMODULATION:NDPSK:INDEX 4 sets the nDPSK number of symbols to 4 symbols for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:DMODULATION:NDPSK:INDEX? may return 8.0000000000, indicating that the number of DPSK symbols is set to 8 for the carrier at index 3.
RFGSignal:CARRier[n]:DMODulation:NDPSK:PROTation

This command sets or returns the Phase Rotation of nDPSK modulation for the specified carrier in the carrier table.

Group  Digital modulation

Syntax  RFGSignal:CARRier[n]:DMODulation:NDPSK:PROTation <rotation>
        RFGSignal:CARRier[n]:DMODulation:NDPSK:PROTation?

Related Commands  RFGSignal:CARRier[n]:DMODulation:TYPE

Arguments  [n]::= <NR1> (“n” determines the carrier index number in the carrier table).
            If omitted, n is interpreted as 1.
            <rotation>::= <NR1> value.

Returns  A single <NR2> value.

Examples  RFGSIGNAL:CARRIER1:DMODULATION:NDPSK:PROTATION 45 sets the nDPSK phase rotation to 45 degrees for the carrier at index 1 in the carrier table.
          RFGSIGNAL:CARRIER3:DMODULATION:NDPSK:PROTATION? might return 180.0000000000, indicating that the nDPSK Phase Rotation is set to 180 degrees for the carrier at index 3.
RFGSignal:CARRier[n]:DMODulation:PSK

This command sets or returns the PSK modulation type for the specified carrier in the carrier table.

**Group**  
Digital modulation

**Syntax**  
RFGSignal:CARRier[n]:DMODulation:PSK {QPSK|PI2QPSK|PI4QPSK|PI4DQPSK|OQPSK|BPSK|NDPSK|PSK8|PI2PSK8|OPSK8|SDPSK|SBPSK|SOQPSK|DQPSK}  
RFGSignal:CARRier[n]:DMODulation:PSK?

**Arguments**  
[n]:= <NR1> (“n” determines the carrier index number in the carrier table).  
If omitted, n is interpreted as 1.

QPSK, PI2QPSK, PI4QPSK, PI4DQPSK, OQPSK, BPSK, NDPSK, PSK8, PI2PSK8, OPSK8, SDPSK, SBPSK, SOQPSK, DQPSK are the selectable PSK digital modulation types.

**Returns**  
QPSK, PI2QPSK, PI4QPSK, PI4DQPSK, OQPSK, BPSK, NDPSK, PSK8, PI2PSK8, OPSK8, SDPSK, SBPSK, SOQPSK, DQPSK.

**Examples**  
RFGSIGNAL:CARRIER1:DMODULATION:PSK PI2QPSK sets the PSK Modulation type to \( \pi/2 \) QPSK for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:DMODULATION:PSK? might return PI2QPSK, indicating that the PSK Modulation type is set to \( \pi/2 \) QPSK for the carrier at index 3.
**RFGSignal:CARRier[n]:DMODulation:QAM**

This command sets or returns the QAM Modulation type for the specified carrier in the carrier table.

**Group**  
Digital modulation

**Syntax**  
RFGSignal:CARRier[n]:DMODulation:QAM {QAM16|PI2QAM16|QAM32  
|QAM64|QAM128|QAM256|QAM512|QAM1024}  
RFGSignal:CARRier[n]:DMODulation:QAM?

**Arguments**  
[n]:= <NR1> (“n” determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

QAM16, PI2QAM16, QAM32, QAM64, QAM128, QAM256, QAM512, and QAM1024 are the QAM modulation types.

**Returns**  
QAM16, PI2QAM16, QAM32, QAM64, QAM128, QAM256, QAM512, or QAM1024 indicating the QAM modulation type.

**Examples**  
RFGSIGNAL:CARRIER1:DMODULATION:QAM QAM32 sets the QAM modulation type to 32 QAM for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:DMODULATION:QAM? might return QAM32, indicating that the QAM Modulation type is set to 32 QAM for the carrier at index 3.
RFGSignal:CARRier[n]:DMODulation:SRATe

This command sets or returns the Digital Modulation Symbol Rate for the specified carrier in the carrier table.

**Group**  
Digital modulation

**Syntax**  
RFGSignal:CARRier[n]:DMODulation:SRATe <rate>  
RFGSignal:CARRier[n]:DMODulation:SRATe?

**Arguments**  
[n]:= <NR1> ("n" determines the carrier index number in the carrier table).  
If omitted, n is interpreted as 1.  
<rate>:= <NR3> value.

**Returns**  
A single <NR3> value.

**Examples**  
RFGSIGNAL:CARRIER1:DMODULATION:SRATE 2E9 sets the Symbol Rate to 2 GHz for the carrier at index 1 in the carrier table.  
RFGSIGNAL:CARRIER3:DMODULATION:SRATE? might return 10.0000000000E+6, indicating that the Symbol Rate is set to 10 MHz for the carrier at index 3
**RFGSignal:CARRier[n]:DMODulation:TYPe**

This command sets or returns the digital modulation type for the specified carrier in the carrier table.

**Group**
Digital modulation

**Syntax**

RFGSignal:CARRier[n]:DMODulation:TYPe

{GMSK|ASK|OOK|CPM|PSK|APSK|QAM|FSK}

RFGSignal:CARRier[n]:DMODulation:TYPe?

**Related Commands**

RFGSignal:CARRier[n]:DMODulation:PSK
RFGSignal:CARRier[n]:DMODulation:APSK
RFGSignal:CARRier[n]:DMODulation:QAM
RFGSignal:CARRier[n]:DMODulation:FSK:PDEViation

**Arguments**

[n]::= <NR1> (“n” determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

GMSK, ASK, OOK, CPM, PSK, APSK, QAM, and FSK are selectable digital modulation types.

PSK, APSK, QAM, and FSK require further definition. See related commands.

**Returns**

GMSK, ASK, OOK, CPM, PSK, APSK, QAM, or FSK indicating the digital modulation type.

**Examples**

RFGSIGNAL:CARRIER1:DMODULATION:TYPE OOK sets the digital modulation type to OOK for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:DMODULATION:TYPE? might return OOK, indicating that the digital modulation type is set to OOK for the carrier at index 3.
RFGSignal:CARRier[n]:FILTER:ALPHA

This command sets or returns the Alpha (filter roll off) value for the specified carrier in the carrier table.

**Conditions**
Filter type must be set to Raised Cosine, Root Raised Cosine, or Gaussian.

**Group**
Filter

**Syntax**
RFGSignal:CARRier[n]:FILTER:ALPHA <alpha>
RFGSignal:CARRier[n]:FILTER:ALPHA?

**Related Commands**
RFGSignal:CARRier[n]:FILTER:TYPE

**Arguments**
[n]:= <NR1> ("n" determines the carrier index number in the carrier table).
If omitted, n is interpreted as 1.

<alpha>:= <NR3> value.

**Returns**
A single <NR3> value.

**Examples**
RFGSIGNAL:CARRIER1:FILTER:ALPHA 0.23 sets the Alpha filter roll off value to 0.23 for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:FILTER:ALPHA? may return 230.0000000000E-3, indicating that the Alpha filter roll off value is set to 0.23 for the carrier at index 3.
**RFGSignal:CARRier[n]:FILTER:CLENght**

This command sets or returns the Filter Convolution Length in symbols for the specified carrier in the carrier table.

**Group**  
Filter

**Syntax**  
RFGSignal:CARRier[n]:FILTER:CLENght <length>  
RFGSignal:CARRier[n]:FILTER:CLENght?

**Arguments**  
[n]:= <NR1> (“n” determines the carrier index number in the carrier table).  
If omitted, n is interpreted as 1.  
<length>:= <NR3> value.

**Returns**  
A single <NR1> value.

**Examples**  
RFGSIGNAL:CARRIER1:FILTER:CLENNGTH 21 sets the Convolution Length to 21 symbols for the carrier at index 1 in the carrier table.  
RFGSIGNAL:CARRIER3:FILTER:CLENNGTH? might return 21.000000000, indicating that the Convolution Length is set to 21 symbols for the carrier at index 3.
**RFGSignal:CARRier[n]:FILTER:FILE**

This command sets or returns the path of the filter file for user defined filter for the specified carrier in the carrier table.

**Group**
Filter

**Syntax**
- `RFGSignal:CARRier[n]:FILTER:FILE <filepath>`
- `RFGSignal:CARRier[n]:FILTER:FILE?`

**Arguments**
- `[n]::= <NR1>` ("n" determines the carrier index number in the carrier table).
  - If omitted, n is interpreted as 1.
- `<filepath>::= <string>`.

**Returns**
A single `<filepath>` string.

**Examples**
- `RFGSignal:CARRier1:FILTER:FILE "c:\test\filter_file.txt"` sets the filter file name to c:\test\filter_file.txt for the carrier at index 1 in the carrier table.
- `RFGSignal:CARRier3:FILTER:FILE?` might return "c:\test\filter_file.txt" for the filter file for the carrier at index 3.
RFGSignal:CARRier[n]:FILTER:TYPE

This command sets or returns the Filter Type for the specified carrier in the carrier table.

Available filters is dependent on the Modulation type selected.

Group Filter

Syntax RFGSignal:CARRier[n]:FILTER:TYPE
{RECTangular|RCOSine|RRCosine|GAUSsian|TRIAngular|EDGe|HSINe |SBPSKMIL|SBPSKSINE|SOQPSKMIL|SOQPSKTG|SOQPSKA|SOQPSKB |UDEFined}
RFGSignal:CARRier[n]:FILTER:TYPE?

Arguments [n]:= <NR1> ("n" determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

RECTangular, RCOSine, RRCosine, GAUSsian, TRIAngular, EDGe, HSINe, and UDEFined are the selectable filter types.

SBPSKMIL and SBPSKSINE are only available when the Modulation is set to SBPSK.

SOQPSKMIL, SOQPSKTG, SOQPSKA, and SOQPSKB are only available when the Modulation is set to SOQPSK.

When setting to UDEF (User Defined), use the command RFGSignal:CARRier[n]:FILTER:FILE to set the path to the user defined filter file.

Returns RECT, RCOS, RRC, GAUS, TRIA, EDG, HSIN, SBPSKMIL, SBPSKSINE, SOQPSKMIL, SOQPSKTG, SOQPSKA, SOQPSKB, or UDEF as the filter type.

Examples RFGSIGNAL:CARRIER1:FILTER:TYPE RRCOSINE sets the Filter Type to Root Raised Cosine for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:FILTER:TYPE? might return RRC, indicating that the Filter Type is set to Root Raised Cosine for the carrier at index 3.
**RFGSignal:CARRier[n]:FILTER:WINDOW:CRIPple**

This command sets or returns the Chebyshev Ripple value when the Filter Window type is Chebyshev Ripple for the specified carrier in the carrier table.

**Group** Filter

**Syntax**

```
RFGSignal:CARRier[n]:FILTER:WINDOW:CRIPple <ripple>
RFGSignal:CARRier[n]:FILTER:WINDOW:CRIPple?
```

**Arguments**

[n]::= <NR1> (“n” determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

<ripple>::=<NR3> value.

**Returns** A single <NR2> value.

**Examples**

```
RFGSIGNAL:CARRIER1:FILTER:WINDOW:CRIPPLE 12 sets the Chebyshev Ripple value to 12 dB for the carrier at index 1 in the carrier table.
```

```
RFGSIGNAL:CARRIER3:FILTER:WINDOW:CRIPPLE? might return 10.0000000000, indicating that the Chebyshev Ripple value for the carrier at index 3 is set to is 10 dB.
```
RFGSignal:CARRier[n]:FILTER:WINDOW:KAISer

This command sets or returns the Kaiser Parameter when the Filter Window type is Kaiser for the specified carrier in the carrier table.

Group       Filter

Syntax       RFGSignal:CARRier[n]:FILTER:WINDOW:KAISer <parameter>
             RFGSignal:CARRier[n]:FILTER:WINDOW:KAISer?

Arguments    [n]::= <NR1> (“n” determines the carrier index number in the carrier table).
              If omitted, n is interpreted as 1.
              <parameter>::= <NR3> value.

Returns      A single <NR3> value.

Examples     RFGSIGNAL:CARRIERS1:FILTER:WINDOW:KAISER 2 sets the Kaiser Parameter to 2 dB for the carrier at index 1 in the carrier table.
             RFGSIGNAL:CARRIERS3:FILTER:WINDOW:KAISER? might return 2, indicating that the Kaiser parameter for the carrier at index 3 is set to 2 dB.
RFGSignal:CARRier[n]:FILTER:WINDow:TYPE

This command sets or returns the Filter Window type for the specified carrier in the carrier table.

**Group**  
Filter

**Syntax**  
RFGSignal:CARRier[n]:FILTER:WINDow:TYPE {NONE|TRIAgular |HAMMing|HANNing|BLACkman|KAISer|BHARris|EBLackman |FTOP|TCOSine|CRIPple}
RFGSignal:CARRier[n]:FILTER:WINDow:TYPE?

**Arguments**  
[n]:= <NR1> (“n” determines the carrier index number in the carrier table).
If omitted, n is interpreted as 1.

- NONE = None
- TRIAgular = Triangular
- HAMMing = Hamming
- HANNing = Hanning
- BLACkman = Blackman
- KAISer = Kaiser
- BHARris = Blackman-Harris
- EBLackman = Exact Blackman
- FTOP = Flat Top
- TCOSine = Tapered Cosine
- CRIPple = Chebyshev Ripple

**Returns**  
NON, TRI, HAMM, HANN, BLAC, KAIS, BHAR, EBL, FTOP, TCOS, CRIP

**Examples**  
RFGSIGNAL:CARRIER1:FILTER:WINDOW:TYPE CRIPPLE sets the Filter Window type to Chebyshev Ripple for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:FILTER:WINDOW:TYPE? might return CRIP, indicating that the Filter Window type for the carrier at index 3 is set to Chebyshev Ripple.
**RFGSignal:CARRier[n]:FREQuency**

This command sets or returns the Carrier Frequency for the specified carrier in the carrier table.

**Group**  
Carrier setup

**Syntax**  
RFGSignal:CARRier[n]:FREQuency <frequency>  
RFGSignal:CARRier[n]:FREQuency?

**Arguments**  
[n]::= <NR1> (“n” determines the carrier index number in the carrier table).  
If omitted, n is interpreted as 1.  
<frequency>::=<NR3> value.

**Returns**  
A single <NR3> value.

**Examples**  
RFGSIGNAL:CARRIER1:FREQUENCY 2E9 sets the Carrier Frequency to 2 GHz for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER2:FREQUENCY? might return 1.0000000000E+9, indicating that the Carrier Frequency for the carrier at index 2 is set to 1 GHz.
RFGSignal:Carrier[n]:HOPping:CUSTom:RLIst

This command sets or returns the Repeat List state (enabled or disabled) for the Frequency Hop List for the specified carrier in the carrier table. When enabled, the Frequency Hop List repeats when the Hop List has been completed.

The Hopping Pattern must be set to Custom and the Hop Time must be set to "Symbols Per Hop", "Hops Per Second", or "Hop Duration".

**Group**  
Hopping

**Syntax**  
RFGSignal:Carrier[n]:HOPping:CUSTom:RLIst {1|0|ON|OFF}  
RFGSignal:Carrier[n]:HOPping:CUSTom:RLIst?

**Related Commands**  
RFGSignal:Carrier[n]:HOPping:PATTern  
RFGSignal:Carrier[n]:HOPping:TIME

**Arguments**  
[n]:= <NR1> (“n” determines the carrier index number in the carrier table).  
If omitted, n is interpreted as 1.

OFF or 0 disables Repeat List. OFF or 0 is the default value.

ON or 1 enables Repeat List.

**Returns**  
A single <Boolean> value.

**Examples**  
RFGSIGNAL:CARRIER1:HOPPING:CUSTOM:RLIST 1 enables the Repeat List so that the Frequency Hop List repeats when the Hop List has been completed for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:HOPPING:CUSTOM:RLIST? might return 0, indicating that the Repeat List is disabled so that the Frequency Hop List is not repeated for the carrier at index 3.
RFGSignal:CARRier[n]:HOPPing[:LIST]:ADD (No Query Form)

This command adds a single entry to the end of the Frequency Hop List or the Frequency Avoid List for the specified carrier in the carrier table.

When the Hopping Pattern is either Custom or Pseudo Random List, the Frequency Hop List modified. When the Hopping Pattern is set to Pseudo Random Range, the Frequency Avoid List is modified.

Group  
Hopping

Syntax  
RFGSignal:CARRier[n]:HOPPing[:LIST]:ADD

Related Commands  
RFGSignal:CARRier[n]:HOPPing:PATTern

Arguments  
[n]:= <NR1> (“n” determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

Examples  
RFG SIGNAL:CARRIER1:HOPPING:LIST:ADD adds a single entry to current hopping list for the carrier at index 1 in the carrier table.
**RFGSignal:CARRier[n]:HOPPing[:LIST]:AOFFset**

This command sets or returns the Amplitude Offset of the currently selected hop for the specified carrier in the carrier table.

The Hopping Pattern must be set to either Custom or Pseudo Random List.

**Group**
Hopping

**Syntax**

\[
\text{RFGSignal:CARRier[n]:HOPPing[:LIST]:AOFFset <offset>}
\]

\[
\text{RFGSignal:CARRier[n]:HOPPing[:LIST]:AOFFset?}
\]

**Related Commands**

- RFGSignal:CARRier[n]:HOPPing:PATTern
- RFGSignal:CARRier[n]:HOPPing[:LIST]:SELect

**Arguments**

\[n::= <NR1> \text{ ("n" determines the carrier index number in the carrier table).} \]

If omitted, n is interpreted as 1.

\[<\text{offset}>::= <\text{NR3}> \text{ value.} \]

**Returns**

A single <NR2> value.

**Examples**

\[
\text{RFGSIGNAL:CARRIER1:HOPPING:LIST:AOFFSET } -10 \text{ sets the Amplitude Offset of the currently selected hop to } -10 \text{ dB for the carrier at index 1 in the carrier table.}
\]

\[
\text{RFGSIGNAL:CARRIER3:HOPPING:LIST:AOFFSET? \ might return } -15.0000000000, \text{ indicating that the Amplitude Offset of the currently selected hop is set to } -15 \text{ dB for the carrier at index 3.}
\]
**RFGSignal:CARRier[n]:HOPping[:LIST]:COUNt? (Query Only)**

This command returns the number of hops in the Hop List or the number of entries in the Frequency Avoid List for the specified carrier in the carrier table.

When the Hopping Pattern is either Custom or Pseudo Random List, the Frequency Hop List count is returned. When the Hopping Pattern is set to Pseudo Random Range, the Frequency Avoid List count is returned.

**Group**

Hopping

**Syntax**

RFGSignal:CARRier[n]:HOPping[:LIST]:COUNt?

**Related Commands**

RFGSignal:CARRier[n]:HOPping:PATTern

**Arguments**

[n]:= <NR1> (“n” determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

**Returns**

A single <NR1> value.

**Examples**

RFGSIGNAL:CARRIER1:HOPPING:LIST:COUNT? might return 15, indicating that there are 15 entries in the list for the carrier at index 1 in the carrier table.
RFGSignal:CARRier[n]:HOPPing[:LIST]:DELete (No Query Form)

This command removes all entries within the Frequency Hop List or the Frequency Avoid List for the specified carrier in the carrier table.

When the Hopping Pattern is either Custom or Pseudo Random List, the Frequency Hop List modified. When the Hopping Pattern is set to Pseudo Random Range, the Frequency Avoid List is modified.

**Group**
Hopping

**Syntax**
RFGSignal:CARRier[n]:HOPPing[:LIST]:DELeTe \{ALL|<index>\}

**Related Commands**
RFGSignal:CARRier[n]:HOPPing:PATTern

**Arguments**
[n]\::= <NR1> (”n” determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

ALL – Deletes all entries of the current hopping list.

<index>\::= <NR1> value. Specifies the index number to delete from the current hopping list.

**Examples**
RFGSIGNAL:CARRIER1:HOPPING:LIST:DELETE ALL removes all entries from the current hopping list for the carrier at index 1 in the carrier table.
**RFGSignal:CARRier[n]:HOPPing[:LIST]:FOFFset**

This command sets or returns the Frequency Offset (Relative Frequency) of the currently selected hop for the specified carrier in the carrier table.

The Hopping Pattern must be set to either Custom or Pseudo Random List.

**Group**  
Hopping

**Syntax**  
RFGSignal:CARRier[n]:HOPPing[:LIST]:FOFFset <offset>  
RFGSignal:CARRier[n]:HOPPing[:LIST]:FOFFset?

**Related Commands**  
RFGSignal:CARRier[n]:HOPPing:PATTern  
RFGSignal:CARRier[n]:HOPPing[:LIST]:SESelect

**Arguments**  
[n]:= <NR1> (“n” determines the carrier index number in the carrier table).  
If omitted, n is interpreted as 1.  
<offset>:= <NR3> value.

**Returns**  
A single <NR3> value.

**Examples**  
RFGSIGNAL:CARRIER1:HOPPING:LIST:FOFFSET 1E9 sets the Frequency Offset of the currently selected hop to 1 GHz for the carrier at index 1 in the carrier table.  
RFGSIGNAL:CARRIER3:HOPPING:LIST:FOFFSET? might return 100.0000000000E+6, indicating that the Frequency Offset of the currently selected hop is set to 100 MHz for the carrier at index 3.
RFGSignal:CARRier[n]:HOPPing[:LIST]:FREQuency

This command sets or returns the Frequency of the currently selected hop in the Frequency Hop List for the specified carrier in the carrier table.

The Hopping Pattern must be set to either Custom or Pseudo Random List.

Group  Hopping

Syntax  
RFGSignal:CARRier[n]:HOPPing[:LIST]:FREQuency <frequency>
RFGSignal:CARRier[n]:HOPPing [:LIST]:FREQuency?

Related Commands  
RFGSignal:CARRier[n]:HOPPing:PA TTern
RFGSignal:CARRier[n]:HOPPing[:LIST]:SELect

Arguments  
[n]:= <NR1> (“n” determines the carrier index number in the carrier table).
If omitted, n is interpreted as 1.

<frequency>:= <NR3> value.

Returns  
A single <NR3> value.

Examples  
RFGSIGNAL:CARRIER1:HOPPING:LIST:FREQUENCY 1E9 sets the Frequency of the currently selected hop to 1 GHz for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:HOPPING:FREQUENCY? might return 100.0000000000E+6, indicating that the Frequency of the currently selected hop is set to 100 MHz for the carrier at index 3.
RFGSignal:CARRier[n]:HOPPing[:LIST]:HDURation

This command sets or returns the Hop Duration of the currently selected hop for the specified carrier in the carrier table.

The Hopping Pattern must set to Custom.

The Hop Time must be set to Hop Duration.

Group          Hopping
Syntax
RFGSignal:CARRier[n]:HOPPing[:LIST]:HDURation <duration>
RFGSignal:CARRier[n]:HOPPing[:LIST]:HDURation?

Related Commands
RFGSignal:CARRier[n]:HOPPing:PATTern
RFGSignal:CARRier[n]:HOPPing:TIME
RFGSignal:CARRier[n]:HOPPing[:LIST]:SELect

Arguments
[n]:= <NR1> ("n" determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

<duration>:= <NR3> value.

Returns
A single <NR3> value.

Examples
RFGSIGNAL:CARRIER1:HOPPING:LIST:HDURATION 1E-6 sets the Hop Duration of the currently selected hop to 1 μs for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:HOPPING:LIST:HDURATION? might return 10.0000000000E-6, indicating that the Hop Duration of the currently selected hop is set to 10 μs for the carrier at index 3.
RFGSignal:CARRier[n]:HOP Ping[:LIST]:INSert (No Query Form)

This command inserts a single entry within the Frequency Hop List or the Frequency Avoid List for the specified carrier in the carrier table. The selected entry point must already exist in the table, otherwise an error is returned.

When the Hopping Pattern is either Custom or Pseudo Random List, the Frequency Hop List modified. When the Hopping Pattern is set to Pseudo Random Range, the Frequency Avoid List is modified.

Group Hopping

Syntax RFGSignal:CARRier[n]:HOPPing[:LIST]:INSert <insertion>

Related Commands RFGSignal:CARRier[n]:HOPPing:PATTern

Arguments [n]::= <NR1> (“n” determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

<insertion>::= <NR1> value.

Examples RFGSIGNAL:CARRIER1:HOPPING:LIST:INSERT 10 inserts a single entry at the 10th index of the current hopping list for the carrier at index 1 in the carrier table.
**RFGSignal:CARRier[n]:HOPPing[:LIST]:SEl ect**

This command sets or returns the selected Hop number in the Frequency Hop List or the Index number in the Frequency Avoid List for the specified carrier in the carrier table.

When the Hopping Pattern is either Custom or Pseudo Random List, the Frequency Hop List is used for selection. When the Hopping Pattern is set to Pseudo Random Range, the Frequency Avoid List is used for selection.

**Group**  
Hopping

**Syntax**  
RFGSignal:CARRier[n]:HOPPing[:LIST]:SEl ect <index>  
RFGSignal:CARRier[n]:HOPPing[:LIST]:SEl ect?

**Related Commands**  
RFGSignal:CARRier[n]:HOPPing:PA TTern

**Arguments**  
[n]::= <NR1> (“n” determines the carrier index number in the carrier table).  
If omitted, n is interpreted as 1.  
<index>::= <NR1> value.

**Returns**  
A single <NR1> value.

**Examples**  
RFGSIGNAL:CARRIER1:HOPPING:LIST:SElECT  3 selects index 3 in the hopping table (Hop list or Avoid list) for the carrier at index 1 in the carrier table.  
RFGSIGNAL:CARRIER3:HOPPING:LIST:SElECT? might return 1, indicating that index 1 in the hopping table (Hop list or Avoid list) is currently selected for modification for the carrier at index 3.
RFGSignal:CARRier[n]:HOPPIng[:LIST]:SSINdex

This command sets or returns the Symbol Start Index (Symbol Index) of the currently selected hop for the specified carrier in the carrier table.

The Hopping Pattern must set to Custom.

The Hop Time must be set to Symbol Start Index.

Group  Hopping

Syntax  RFGSignal:CARRier[n]:HOPPIng[:LIST]:SSINdex <start_index>
        RFGSignal:CARRier[n]:HOPPIng[:LIST]:SSINdex?

Related Commands  RFGSignal:CARRier[n]:HOPPIng:PATTern
                   RFGSignal:CARRier[n]:HOPPIng:TIME
                   RFGSignal:CARRier[n]:HOPPIng[:LIST]:SELect

Arguments  [n]:= <NR1> (“n” determines the carrier index number in the carrier table).
           If omitted, n is interpreted as 1.
           <start_index>:= <NR1> value.

Returns  a single <NR1> value.

Examples  RFGSIGNAL:CARRIER1:HOPPING:LIST:SSINDEX 10 sets the Symbol Start Index of the currently selected hop to symbol 10 for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:HOPPING:SSINDEX? might return 6, indicating that the Symbol Start Index of the currently selected hop is set to 6 for the carrier at index 3.
**RFGSignal:CARRier[n]:HOPPing:PATTern**

This command sets or returns the Hopping Pattern for the specified carrier in the carrier table.

**Group**  
Hopping

**Syntax**  
RFGSignal:CARRier[n]:HOPPing:PATTern \{CUSTom|LIST|RANGe}\  
RFGSignal:CARRier[n]:HOPPing:PATTern?

**Arguments**  
[n]::= <NR1> (“n” determines the carrier index number in the carrier table).  
If omitted, n is interpreted as 1.  
CUSTom = Custom hopping pattern.  
LIST = Pseudo Random List hopping pattern.  
RANGe = Pseudo Random Range hopping pattern.

**Returns**  
CUST  
LIST  
RANG

**Examples**  
RFGSIGNAL:CARRIER1:HOPPING:PATTERN LIST sets the Hopping Pattern to be Pseudo Random List for the carrier at index 1 in the carrier table.  
RFGSIGNAL:CARRIER3:HOPPING:PATTERN? might return RANG, indicating that the Hopping Pattern is set to Pseudo Random Range for the carrier at index 3.
**RFGSignal:CARRier\[n\]:HOPPing:PRBS**

This command sets or returns the Hopping PRBS pattern for the specified carrier in the carrier table.

The Hopping Pattern must be set to Pseudo Random List or Pseudo Random Range.

**Group**

Hopping

**Syntax**

```
RFGSignal:CARRier\[n\]:HOPPing:PRBS
{PRBS7|PRBS9|PRBS15|PRBS16|PRBS20|PRBS21|PRBS23|PRBS29
|PRBS31|UDEFined}
```

```
RFGSignal:CARRier\[n\]:HOPPing:PRBS?
```

**Related Commands**

RFGSignal:CARRier\[n\]:HOPPing:PATTern

**Arguments**

\[n\]: ::= <NR1> (“n” determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

PRBS7, PRBS9, PRBS15, PRBS16, PRBS20, PRBS21, PRBS23, PRBS29, PRBS31, and UDEF are the PRBS types.

When setting to UDEF (User Defined), use the commands

RFGSignal:CARRier\[n\]:HOPPing:PRBS:UDEFined:POLYnomial and
RFGSignal:CARRier\[n\]:HOPPing:PRBS:UDEFined:SREGister
to define the PRBS Polynomial Expression and Shift register initial value.

**Returns**

PRBS7, PRBS9, PRBS15, PRBS16, PRBS20, PRBS21, PRBS23, PRBS29, PRBS31, UDEF

**Examples**

RFGSIGNAL:CARRIER1:HOPPING:PRBS PRBS21 sets the Hopping PRBS to PRBS 21 for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:HOPPING:PRBS? might return PRBS15, indicating that the Hopping PRBS is set to PRBS15 for the carrier at index 3.
RFGSignal:CARRier[n]:HOPPing:PRBS:UDEFined:POLYnomial

This command sets or returns the Hopping PRBS Polynomial Expression for the User Defined PRBS for the specified carrier in the carrier table.

The Hopping Pattern must be set to Pseudo Random List or Pseudo Random Range.

**Group**  
Hopping

**Syntax**  
RFGSignal:CARRier[n]:HOPPing:PRBS:UDEFined:POLYnomial

<polynomial>

RFGSignal:CARRier[n]:HOPPing:PRBS:UDEFined:POLYnomial?

**Related Commands**  
RFGSignal:CARRier[n]:HOPPing:PATTern

**Arguments**  
[n]:= <NR1> ("n" determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

<polynomial>::= <string>.

The highest degree of polynomial is 31. If the expression contains more than 31 an error will be returned. For example X32+X23+1 returns an error.

**Returns**  
A single <polynomial> string.

**Examples**  
RFGSIGNAL:CARRIER1:HOPPING:PRBS:UDEFINED:POLYNOMIAL

"X12+X11+1" sets the Hopping User Defined PRBS polynomial expression to X12+X11+1 for the carrier 2.

RFGSIGNAL:CARRIER3:HOPPING:PRBS:UDEFINED:POLYNOMIAL? might return "X12+X11+1", indicating this is the Hopping User Defined PRBS polynomial for the carrier at index 3.
RFGSignal:CARRier[n]:HOPPing:PRBS:UDEFined:RESet (No Query Form)

This command resets the Hopping User Defined PRBS type to its default value for the specified carrier in the carrier table.

The Hopping Pattern must be set to Pseudo Random List or Pseudo Random Range.

Group  Hopping

Syntax  RFGSignal:CARRier[n]:HOPPing:PRBS:UDEFined:RESet

Arguments  [n]:= <NR1> (“n” determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

Examples  RFGSIGNAL:CARRIER1:HOPPING:PRBS:UDEFINED:RESET resets the Hopping User Defined PRBS type for the carrier at index 1 in the carrier table.
**RFGSignal:CARRier[n]:HOPPIng:PRBS:UDEFined:SREGister**

This command sets or returns the Hopping PRBS Polynomial Expression for the User Defined PRBS for the specified carrier in the carrier table.

The Hopping Pattern must be set to Pseudo Random List or Pseudo Random Range.

**Group**  
Hopping

**Syntax**

```plaintext
RFGSignal:CARRier[n]:HOPPing:PRBS:UDEFined:SREGister
<polynomial>
RFGSignal:CARRier[n]:HOPPing:PRBS:UDEFined:SREGister?
```

**Related Commands**

`RFGSignal:CARRier[n]:HOPPing:PATTern`

**Arguments**

`[n]::= <NR1>` ("n" determines the carrier index number in the carrier table).

  If omitted, n is interpreted as 1.

`<polynomial>::= <string>`.

  The highest degree of polynomial is 31. If the expression contains more than 31 an error will be returned. For example X32+X23+1 returns an error.

**Returns**

A single `<polynomial>` string.

**Examples**

`RFGSIGNAL:CARRIER1:HOPPING:PRBS:UDEFINED:SREGISTER "11011"` sets the Hopping User Defined PRBS Shift register to 11011 for the carrier at index 1 in the carrier table.

RFGSignal:CARRier[n]:HOPPing:RANGe:ALISt[:ENABLE]

This command sets or returns the Frequency Avoid List state (enabled or disabled) for the Pseudo Random Range Hopping Pattern for the specified carrier in the carrier table.

The Hopping Pattern must be set to Pseudo Random Range.

Group
Hopping

Syntax
RFGSignal:CARRier[n]:HOPPing:RANGe:ALISt[:ENABLE]
{1|0|ON|OFF}
RFGSignal:CARRier[n]:HOPPing:RANGe:ALISt[:ENABLE]?

Related Commands
RFGSignal:CARRier[n]:HOPPing:PATTern

Arguments
[n]:= <NR1> (“n” determines the carrier index number in the carrier table).
If omitted, n is interpreted as 1.
OFF or 0 disables the Frequency Avoid List. OFF or 0 is the default value.
ON or 1 enables the Frequency Avoid List.

Returns
A single <Boolean> value.

Examples
RFGSIGNAL:CARRIER1:HOPPING:RANGE:ALIST:ENABLE 1 enables the Frequency Avoid List for the carrier at index 1 in the carrier table.
RFGSIGNAL:CARRIER3:HOPPING:RANGE:ALIST:ENABLE? might return 0, indicating that the Frequency Avoid List is disabled for the carrier at index 3.
**RFGSignal:CARRier[n]:HOPPing:RANGe[:FREQuency]:MAXimum**

This command sets or returns the Maximum Frequency value for the Pseudo Random Range Hopping Pattern for the specified carrier in the carrier table.

The Hopping Pattern must be set to Pseudo Random Range.

**Group**

Hopping

**Syntax**

RFGSignal:CARRier[n]:HOPPing:RANGe[:FREQuency]:MAXimum

<frequency>

RFGSignal:CARRier[n]:HOPPing:RANGe[:FREQuency]:MAXimum?

**Related Commands**

RFGSignal:CARRier[n]:HOPPing:ATTERn

**Arguments**

[n]:= <NR1> (“n” determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

<frequency>:= <NR3> value.

**Returns**

A single <NR3> value.

**Examples**

RFGSIGNAL:CARRIER1:HOPPING:RANGE:FREQUENCY:MAXIMUM 2E9 sets the Pseudo Random Range Maximum Frequency to 2 GHz for the carrier at index 1 in the carrier table.

**RFGSignal:CARRier[n]:HOPPing:RANGe[:FREQuency]:MINimum**

This command sets or returns the Minimum Frequency value for the Pseudo Random Range Hopping Pattern for the specified carrier in the carrier table. The Hopping Pattern must be set to Pseudo Random Range.

**Group**  
Hopping

**Syntax**  
RFGSignal:CARRier[n]:HOPPing:RANGe[:FREQuency]:MINimum  
RFGSignal:CARRier[n]:HOPPing:RANGe[:FREQuency]:MINimum?

**Related Commands**  
RFGSignal:CARRier[n]:HOPPing:PATTern

**Arguments**  
[n]::= <NR1> (“n” determines the carrier index number in the carrier table).  
If omitted, n is interpreted as 1.  
<frequency>::= <NR3> value.

**Returns**  
A single <NR3> value.

**Examples**  
RFGSIGNAL:CARRIER1:HOPPING:RANGE:FREQUENCY:MINIMUM 1E9 sets the Pseudo Random Range Minimum Frequency to 1 GHz for the carrier at index 1 in the carrier table.  
**RFGSignal:CARRier[n]:HOPPing:RANGe[:FREQuency]:SPACing**

This command sets or returns the Frequency Spacing value for the Pseudo Random Range Hopping Pattern for the specified carrier in the carrier table.

The Hopping Pattern must be set to Pseudo Random Range.

**Group**
Hopping

**Syntax**
RFGSignal:CARRier[n]:HOPPing:RANGe[:FREQuency]:SPACing
<spacing>
RFGSignal:CARRier[n]:HOPPing:RANGe[:FREQuency]:SPACing?

**Related Commands**
RFGSignal:CARRier[n]:HOPPing:PATTern

**Arguments**
[n]:= <NR1> (“n” determines the carrier index number in the carrier table).
If omitted, n is interpreted as 1.
<spacing>::= <NR3> value.

**Returns**
A single <NR3> value.

**Examples**
RFGSIGNAL:CARRIER1:HOPPING:RANGE:FREQUENCY:SPACING 1E9 sets the Pseudo Random Range Frequency Spacing to 1 MHz for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER1:HOPPING:RANGE:FREQUENCY:SPACING? might return 10.0000000000E+3, indicating that the Pseudo Random Range Frequency Spacing is set to 100 kHz for the carrier at index 3.
RFGSignal:CARRier[n]:HOPPing:TIME

This command sets or returns the Hop Time type for the specified carrier in the carrier table.

**Group**

Hopping

**Syntax**

RFGSignal:CARRier[n]:HOPping:TIME
{SPHopp|HPSecond|SSINdex|HDURation}
RFGSignal:CARRier[n]:HOPping:TIME?

**Arguments**

[n]:= <NR1> (“n” determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

SPHop = Symbols Per Hop.

HPSecond = Hops Per Second.

SSINdex = Symbol Start Index. (Available for Custom Hopping Pattern.)

HDURation = Hop Duration. (Available for Custom Hopping Pattern.)

**Returns**

SPH

HPS

SSIN

HDUR

**Examples**

RFGSIGNAL:CARRIER1:HOPPING:TIME HPS sets the Hop Time to Hops Per Second for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:HOPPING:TIME? might return SSIN, indicating that the Hop Time is set to Symbol Start Index for the carrier at index 3.
**RFGSignal:CARRier[n]:HOPPing:TIME:HPSecond**

This command sets or returns the Hops per Second for the specified carrier in the carrier table.

The Hop Time must be set to Hops Per Second.

**Group**

Hopping

**Syntax**

RFGSignal:CARRier[n]:HOPping:TIME:HPSecond <hops>

RFGSignal:CARRier[n]:HOPping:TIME:HPSecond?

**Related Commands**

RFGSignal:CARRier[n]:HOPping:TIME

**Arguments**

[n]:= <NR1> (“n” determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

<hops>::= <NR1> value.

**Returns**

A single <NR1> value.

**Examples**

RFGSIGNAL:CARRIER1:HOPPING:TIME:HPSECOND 50 sets the Hops Per Second to 50 for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:HOPPING:TIME:HPSECOND? might return 1000, indicating that the Hops per Second is set to 1000 for the carrier at index 3.
RFGSignal:CARRier[n]:HOPPing:TIME:SPHop

This command sets or returns the Symbols per Hop for the specified carrier in the carrier table.

The Hop Time must be set to Symbols Per Hop.

Group
Hopping

Syntax
RFGSignal:CARRier[n]:HOPPing:TIME:SPHop <symbols>
RFGSignal:CARRier[n]:HOPPing:TIME:SPHop?

Related Commands
RFGSignal:CARRier[n]:HOPPing:TIME

Arguments
[n]::= <NR1> (“n” determines the carrier index number in the carrier table).
   If omitted, n is interpreted as 1.
<symbols>::= <NR1> value.

Returns
A single <NR1> value.

Examples
RFGSIGNAL:CARRIER1:HOPPING:TIME:SPHOP 50 sets the Symbols Per Hop to 50 symbols for the carrier at index 1 in the carrier table.
RFGSIGNAL:CARRIER3:HOPPING:TIME:SPHOP? might return 1200, indicating that the Symbols per Hop is set to 1200 symbols for the carrier at index 3.
RFGSignal:CARRier[n]:HOPPing:TON

This command sets or returns the Hopping state (enabled or disabled) for the specified carrier in the carrier table.

Group
Hopping

Syntax
RFGSignal:CARRier[n]:HOPPing:TON {1|0|ON|OFF}
RFGSignal:CARRier[n]:HOPPing:TON?

Arguments
[n]:= <NR1> ("n" determines the carrier index number in the carrier table).
If omitted, n is interpreted as 1.
OFF or 0 disables Hopping. OFF or 0 is the default value.
ON or 1 enables Hopping.

Returns
A single <Boolean> value.

Examples
RFGSIGNAL:CARRIER1:HOPPING:TON ON enables Hopping for the carrier at index 1 in the carrier table.
RFGSIGNAL:CARRIER3:HOPPING:TON? might return 0, indicating that Hopping is disabled for the carrier at index 3.
RFGSignal:CARRier[n]:INterference:ANOIse:BANDwidth

This command sets or returns the Bandwidth for Additive Noise for Interference Addition for the specified carrier in the carrier table.

Group
   Interference addition

Syntax
   RFGSignal:CARRier[n]:INterference:ANOIse:BANDwidth
   <bandwidth>
   RFGSignal:CARRier[n]:INterference:ANOIse:BANDwidth?

Arguments
   [n]::= <NR1> (“n” determines the carrier index number in the carrier table).
   If omitted, n is interpreted as 1.
   <bandwidth>::= <NR3> value.

Returns
   A single <NR3> value.

Examples
   RFGSIGNAL:CARRIER1:INTERFERENCE:ANOISE:BANDWIDTH 1E6 sets the Additive Noise Bandwidth to 1 MHz for the carrier at index 1 in the carrier table.
   RFGSIGNAL:CARRIER3:INTERFERENCE:ANOISE:BANDWIDTH? might return 8.5000000000E+6, indicating that the Additive Noise Bandwidth is set to 8.5 MHz for the carrier at index 3.
### RFGSignal:CARRier[n]:INTerference:ANOlse:EBNo

This command sets or returns the Eb/No (bit energy per noise power) for Additive Noise for Interference Addition for the specified carrier in the carrier table.

Setting the Eb/No value also selects this as the Additive Noise type.

**Group**  
Interference addition

**Syntax**  
RFGSignal:CARRier[n]:INTerference:ANOlse:EBNo <level>  
RFGSignal:CARRier[n]:INTerference:ANOlse:EBNo?

**Arguments**  
[n]:= <NR1> (“n” determines the carrier index number in the carrier table).  
If omitted, n is interpreted as 1.  
<level>:= <NR3> value.

**Returns**  
A single <NR2> value.

**Examples**  
RFGSIGNAL:CARRIER1:INTERFERENCE:ANOISE:EBNO –5 sets the Eb/No level to –5 dB for the carrier at index 1 in the carrier table.  
RFGSIGNAL:CARRIER3:INTERFERENCE:ANOISE:EBNO? might return 1.2000000000, indicating that the Eb/No level is set to 1.2 dB for the carrier at index 3.
RFGSignal:CARRier[n]:INTerference:ANOIse:SNR

This command sets or returns the SNR for Additive Noise for Interference Addition for the specified carrier in the carrier table.

Setting the SNR value also selects this as the Additive Noise type.

Group  
Interference addition

Syntax  
RFGSignal:CARRier[n]:INTerference:ANOIse:SNR <level>
RFGSignal:CARRier[n]:INTerference:ANOIse:SNR?

Arguments  
[n]::= <NR1> (“n” determines the carrier index number in the carrier table).
If omitted, n is interpreted as 1.
<level>::= <NR3> value.

Returns  
A single <NR3> value.

Examples  
RFGSIGNAL:CARRIER1:INTERFERENCE:ANOISE:SNR 5 sets the SNR level to 5 dB for the carrier at index 1 in the carrier table.

-20.0000000000, indicating that the SNR level is set to -20 dB for the carrier at index 3.
RFGSignal:CARRier[n]:INTERference:ANOise:TON

This command sets or returns the Additive Noise state (enabled or disabled) for Interference Addition for the specified carrier in the carrier table.

Group  
Interference addition

Syntax  
RFGSignal:CARRier[n]:INTERference:ANOise:TON {1|0|ON|OFF}  
RFGSignal:CARRier[n]:INTERference:ANOise:TON?

Arguments  
[n]:= <NR1> (“n” determines the carrier index number in the carrier table).
If omitted, n is interpreted as 1.
OFF or 0 disables Additive Noise. OFF or 0 is the default value.
ON or 1 enables Additive Noise.

Returns  
A single <Boolean> value.

Examples  
RFGSIGNAL:CARRIER1:INTERFERENCE:ANOISE:TON  1 enables the Additive Noise for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:INTERFERENCE:ANOISE:TON? might return 0, indicating that the Additive Noise is disabled for the carrier at index 3.
**RFGSignal:CARRier[n]:INTerference:FOFFset**

This command sets or returns the Frequency Offset frequency for Interference Addition for the specified carrier in the carrier table.

**Group**
Interference addition

**Syntax**
```
RFGSignal:CARRier[n]:INTerference:FOFFset <frequency>
RFGSignal:CARRier[n]:INTerference:FOFFset?
```

**Arguments**
[n]::= <NR1> (“n” determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

<frequency>::= <NR3> value.

**Returns**
A single <NR3> value.

**Examples**
```
RFGSIGNAL:CARRIER1:INTERFERENCE:FOFFSET 1E5 sets the Frequency Offset frequency to 100 kHz for the carrier at index 1 in the carrier table.
```

```
RFGSIGNAL:CARRIER3:INTERFERENCE:FOFFSET? might return 1.0000000000E+6, indicating that the Frequency Offset frequency is set to 1 MHz for the carrier at index 3.
```
RFGSignal:CARRier[n]:INterference:FOFFset:TON

This command sets or returns the Frequency Offset state (enabled or disabled) for Interference Addition for the specified carrier in the carrier table.

Group Interference addition

Syntax

RFGSignal:CARRier[n]:INterference:FOFFset:TON {1|0|ON|OFF}
RFGSignal:CARRier[n]:INterference:FOFFset:TON?

Arguments

[n]:= <NR1> (“n” determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

OFF or 0 disables Frequency Offset. OFF or 0 is the default value.

ON or 1 enables Frequency Offset.

Returns A single <Boolean> value.

Examples

RFGSIGNAL:CARRIER1:INTERERENCE:FOFFSET:TON 1 enables the Frequency Offset for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER1:INTERERENCE:FOFFSET:TON? might return 0, indicating that the Frequency Offset is disabled for the carrier at index 3.
RFGSignal:CARRier[n]:INTerference:SINusoidal:CI

This command sets or returns the C/I level for Sinusoidal Interference for Interference Addition for the specified carrier in the carrier table.

**Group**
Interference addition

**Syntax**
RFGSignal:CARRier[n]:INTerference:SINusoidal:CI <level>
RFGSignal:CARRier[n]:INTerference:SINusoidal:CI?

**Arguments**
[n]::= <NR1> (“n” determines the carrier index number in the carrier table).
If omitted, n is interpreted as 1.

<level>::= <NR3> value.

**Returns**
A single <NR3> value.

**Examples**
RFGSIGNAL:CARRIER1:INTERFERENCE:SINUSOIDAL:CI 10.5 sets the Sinusoidal Interference C/I level to 10.5 dB for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:INTERFERENCE:SINUSOIDAL:CI? might return 3.0000000000, indicating that the Sinusoidal Interference C/I level is set to 3 dB for the carrier at index 3.
**RFGSignal:CARRier[n]:INTerference:SINusoidal:COFFset**

This command sets or returns the Offset from Carrier frequency for Sinusoidal Interference for Interference Addition for the specified carrier in the carrier table.

**Group**
Interference addition

**Syntax**

```
RFGSignal:CARRier[n]:INTerface:SINusoidal:COFFset
<frequency>
RFGSignal:CARRier[n]:INTerface:SINusoidal:COFFset?
```

**Arguments**

[n]:= <NR1> ("n" determines the carrier index number in the carrier table).
- If omitted, n is interpreted as 1.

<frequency>:= <NR3> value.

**Returns**
A single <NR3> value.

**Examples**

```
RFGSIGNAL:CARRIER1:INTERFERENCE:SINUSOIDAL:COFFSET 1E6 sets the Offset from Carrier frequency to 1 MHz for the carrier at index 1 in the carrier table.
```

```
RFGSIGNAL:CARRIER3:INTERFERENCE:SINUSOIDAL:COFFSET? might return 100.0000000000E+3, indicating that the Offset from Carrier frequency is set to 1 MHz for the carrier at index 3.
```
RFGSignal:CARRier[n]:INterference:SINusoidal:TON

This command sets or returns the Sinusoidal Interference state (enabled or disabled) for Interference Addition for the specified carrier in the carrier table.

**Group**  
Interference addition

**Syntax**  
RFGSIGNAL:CARRIER[n]:INTERERENCE:SINUSOIDAL:TON  
{1|0|ON|OFF}  
RFGSIGNAL:CARRIER[n]:INTERERENCE:SINUSOIDAL:TON?

**Arguments**  
[n]::= <NR1> (“n” determines the carrier index number in the carrier table).  
If omitted, n is interpreted as 1.  
OFF or 0 disables Sinusoidal Interference. OFF or 0 is the default value.  
ON or 1 enables Sinusoidal Interference.

**Returns**  
A single <Boolean> value.

**Examples**  
RFGSIGNAL:CARRIER1:INTERERENCE:SINUSOIDAL:TON 1 enables Sinusoidal Interference for the carrier at index 1 in the carrier table.  
RFGSIGNAL:CARRIER3:INTERERENCE:SINUSOIDAL:TON? might return 0, indicating that the Sinusoidal Interference is disabled for the carrier at index 3.
RFGSignal:Carrier[n]:IQAMplitude

This command sets or returns the IQ Amplitude for the specified carrier in the carrier table.

**Conditions**
The signal type must be set to IQ.

**Group**
Carrier setup

**Syntax**
RFGSignal:Carrier[n]:IQAMplitude <amplitude>
RFGSignal:Carrier[n]:IQAMplitude?

**Arguments**
[n]::= <NR1> (“n” determines the carrier index number in the carrier table).
If omitted, n is interpreted as 1.
<amplitude>::=<NR3> value.

**Returns**
A single <NR3> value.

**Examples**
RFGSIGNAL:CARRIER1:IQAMPLITUDE 350E-3 sets the IQ Amplitude to 350 mVrms for the carrier at index 1 in the carrier table.
RFGSIGNAL:CARRIER3:IQAMPLITUDE? might return 150.0000000000E-3, indicating that the IQ Amplitude is set to 150 mVrms for the carrier at index 3.
RFGSignal:CARRier[n]:IQIMpairment:CLEAkage:IOFFset

This command sets or returns the I Offset percentage for Carrier Leakage for IQ Impairments for the specified carrier in the carrier table.

Group  IQ impairments

Syntax  RFGSignal:CARRier[n]:IQIMpairment:CLEAkage:IOFFset
        <percentage>
        RFGSignal:CARRier[n]:IQIMpairment:CLEAkage:IOFFset?

Arguments  [n]::= <NR1> (“n” determines the carrier index number in the carrier table).
            If omitted, n is interpreted as 1.
            <percentage>::= <NR3> value.

Returns  A single <NR3> value.

Examples  RFGSIGNAL:CARRIER1:IQIMPAIRMENT:CLEAKAGE:IOFFSET -10 sets the Carrier Leakage I Offset percentage to –10 % for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:IQIMPAIRMENT:CLEAKAGE:IOFFSET? might return 3.0000000000, indicating that the Carrier Leakage I Offset percentage is set to 3 % for the carrier at index 3.
RFGsignal:CARRier[n]:IQIMpairment:CLEAkage:IQOFfset

This command sets or returns the IQ Offset level for Carrier Leakage for IQ Impairments for the specified carrier in the carrier table.

Group   IQ impairments

Syntax   RFGsignal:CARRier[n]:IQIMpairment:CLEAkage:IQOFfset <level>
           RFGsignal:CARRier[n]:IQIMpairment:CLEAkage:IQOFfset?

Arguments  [n]:= <NR1> (“n” determines the carrier index number in the carrier table).
           If omitted, n is interpreted as 1.
           <level>:= <NR1> value.

Returns    A single <NR2> value.

Examples   RFGSIGNAL:CARRIER1:IQIMPAIRMENT:CLEAKAGE:IQOFFSET -5 sets the Carrier Leakage IQ Offset level to –5 dB for the carrier at index 1 in the carrier table.
**RFGSignal:CARRier[n]:IQIMpairment:CLEAkage:QOFFset**

This command sets or returns the Q Offset percentage for Carrier Leakage for IQ Impairments for the specified carrier in the carrier table.

**Group**  
IQ impairments

**Syntax**  

```
RFGSignal:CARRier[n]:IQIMpairment:CLEAkage:QOFFset
<percentage>
RFGSignal:CARRier[n]:IQIMpairment:CLEAkage:QOFFset?
```

**Arguments**  

- `[n]::= <NR1>` (“n” determines the carrier index number in the carrier table).  
  If omitted, n is interpreted as 1.  
- `<percentage>::= <NR3>` value.

**Returns**  
A single `<NR3>` value.

**Examples**  

```
RFGSIGNAL:CARRIER1:IQIMPAIRMENT:CLEAKAGE:QOFFSET -10 sets the Carrier Leakage Q Offset percentage to –10 % for the carrier at index 1 in the carrier table.
```

```
RFGSIGNAL:CARRIER3:IQIMPAIRMENT:CLEAKAGE:QOFFSET? might return 3.0000000000, indicating that the Carrier Leakage Q Offset percentage is set to 3 % for the carrier at index 3.
```
RFGSignal:CARRier[n]:IQIMpairment:CLEAkage:TON

This command sets or returns the Carrier Leakage state (enabled or disabled) for IQ Impairments for the specified carrier in the carrier table.

**Group**
IQ impairments

**Syntax**
RFGSignal:CARRier[n]:IQIMpairment:CLEAkage:TON {1|0|ON|OFF}
RFGSignal:CARRier[n]:IQIMpairment:CLEAkage:TON?

**Arguments**
[n]:= <NR1> (“n” determines the carrier index number in the carrier table).
If omitted, n is interpreted as 1.
OFF or 0 disables Carrier Leakage. OFF or 0 is the default value.
ON or 1 enables Carrier Leakage.

**Returns**
A single <Boolean> value.

**Examples**
RFGSIGNAL:CARRIER1:IQIMPAIRMENT:CLEAKAGE:TON 1 enables the Carrier Leakage for the carrier at index 1 in the carrier table.
RFGSIGNAL:CARRIER3:IQIMPAIRMENT:CLEAKAGE:TON? might return 0, indicating that the Carrier Leakage is disabled for the carrier at index 3.
**RFGSignal:CARRier[n]:IQIMpairment:IQIMbalance:IMBalance**

This command sets or returns the Imbalance percentage for IQ Imbalance for IQ Impairments for the specified carrier in the carrier table.

**Group**  
IQ impairments

**Syntax**  
RFGSignal:CARRier[n]:IQIMpairment:IQIMbalance:IMBalance
<percentage>
RFGSignal:CARRier[n]:IQIMpairment:IQIMbalance:IMBalance?

**Arguments**  
[n]::= <NR1> (“n” determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

<percentage>::= <NR3> value.

**Returns**  
A single <NR3> value.

**Examples**  
RFGSIGNAL:CARRIER1:IQIMPAIRMENT:IQIMBALANCE:IMBALANCE –11 sets the IQ Imbalance percentage to –11 % for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:IQIMPAIRMENT:IQIMBALANCE:IMBALANCE? might return 3.0000000000, indicating that the IQ Imbalance percentage is set to 3 % for the carrier at index 3.
RFGSignal:CARRIER[n]:IQIMpairment:IQIMbalance:TON

This command sets or returns the IQ Imbalance state (enabled or disabled) for IQ Impairments for the specified carrier in the carrier table.

Group  IQ impairments

Syntax  
RFGSignal:CARRier[n]:IQIMpairment:IQIMbalance:TON
{1|0|ON|OFF}
RFGSignal:CARRier[n]:IQIMpairment:IQIMbalance:TON?

Arguments  
[n]:= <NR1> (“n” determines the carrier index number in the carrier table).
  If omitted, n is interpreted as 1.
  OFF or 0 disables IQ Imbalance. OFF or 0 is the default value.
  ON or 1 enables IQ Imbalance.

Returns  A single <Boolean> value.

Examples  
RFGSIGNAL:CARRIER1:IQIMPAIRMENT:IQIMBALANCE:TON 1 enables the IQ Imbalance for the carrier at index 1 in the carrier table.
RFGSIGNAL:CARRIER3:IQIMPAIRMENT:IQIMBALANCE:TON? might return 0, indicating that the IQ Imbalance is disabled for the carrier at index 3.
RFGSignal:CARRier[n]:IQIMpairment:IQSWap:TON

This command sets or returns the Swap I & Q state (enabled or disabled) for IQ Impairments for the specified carrier in the carrier table.

**Group**  
IQ impairments

**Syntax**  
RFGSignal:CARRier[n]:IQIMpairment:IQSWap:TON {1|0|ON|OFF}  
RFGSignal:CARRier[n]:IQIMpairment:IQSWap:TON?

**Arguments**  
[n]::= <NR1> (“n” determines the carrier index number in the carrier table).  
If omitted, n is interpreted as 1.  
OFF or 0 disables Swap I & Q. OFF or 0 is the default value.  
ON or 1 enables Swap I & Q.

**Returns**  
A single <Boolean> value.

**Examples**  
RFGSIGNAL:CARRIER1:IQIMPAIRMENT:IQSWAP:TON 1 enables Swap I & Q for the carrier at index 1 in the carrier table.  
RFGSIGNAL:CARRIER3:IQIMPAIRMENT:IQSWAP:TON? might return 0, indicating that Swap I & Q is disabled for the carrier at index 3.
RFGSignal:CARRIER[n]:IQIMpairment:NLDistortion:AM2K

This command sets or returns the k2 level for AM/AM Nonlinear Distortions for IQ Impairments for the specified carrier in the carrier table.

Group
IQ impairments

Syntax
RFGSignal:CARRIER[n]:IQIMpairment:NLDistortion:AM2K <level>
RFGSignal:CARRIER[n]:IQIMpairment:NLDistortion:AM2K?

Arguments
[n]::= <NR1> (“n” determines the carrier index number in the carrier table).
If omitted, n is interpreted as 1.
[level]::= <NR3> value.

Returns
A single <NR3> value.

Examples
RFGSIGNAL:CARRIER1:IQIMPAIRMENT:NLDISTORTION:AM2K −1 sets the AM/AM k2 level to −1 dB for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:IQIMPAIRMENT:NLDISTORTION:AM2K? might return −1.0000000000, indicating that the AM/AM k2 level is set to −1 dB for the carrier at index 3.
RFGSignal:CARRier[n]:IQIMpairment:NLDistortion:AM3K

This command sets or returns the k3 level for AM/AM Nonlinear Distortions for IQ Impairments for the specified carrier in the carrier table.

Group
IQ impairments

Syntax
RFGSignal:CARRier[n]:IQIMpairment:NLDistortion:AM3K <level>
RFGSignal:CARRier[n]:IQIMpairment:NLDistortion:AM3K?

Arguments
[n]::= <NR1> ("n" determines the carrier index number in the carrier table).
If omitted, n is interpreted as 1.

<level>::= <NR3> value.

Returns
A single <NR3> value.

Examples
RFGSIGNAL:CARRIER1:IQIMPAIRMENT:NLDISTORTION:AM3K –1 sets the AM/AM k3 level to –1 dB for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:IQIMPAIRMENT:NLDISTORTION:AM3K? might return –1.0000000000, indicating that the AM/AM k3 level is set to –1 dB for the carrier at index 3.
**RFGSignal:CARRier[n]:IQIMpairment:NLDistortion:PM2K**

This command sets or returns the k2 level for AM/PM Nonlinear Distortions for IQ Impairments for the specified carrier in the carrier table.

**Group**  
IQ impairments

**Syntax**  
RFGSignal:CARRier[n]:IQIMpairment:NLDistortion:PM2K <level>  
RFGSignal:CARRier[n]:IQIMpairment:NLDistortion:PM2K?

**Arguments**  
[n]::= <NR1> ("n" determines the carrier index number in the carrier table).  
If omitted, n is interpreted as 1.  
<level> ::= <NR3> value.

**Returns**  
A single <NR3> value.

**Examples**  
RFGSIGNAL:CARRIER1:IQIMPAIRMENT:NLDISTORTION:PM2K -1 sets the AM/PM k2 level to –1 dB for the carrier at index 1 in the carrier table.  
RFGSIGNAL:CARRIER3:IQIMPAIRMENT:NLDISTORTION:PM2K? might return –1.0000000000, indicating that the AM/PM k2 level is set to –1 dB for the carrier at index 3.
**RFGSignal:CARRier[n]:IQIMpairment:NLDistortion:PM3K**

This command sets or returns the k3 level for AM/PM Nonlinear Distortions for IQ Impairments for the specified carrier in the carrier table.

**Group**  
IQ impairments

**Syntax**  
RFGSignal:CARRier[n]:IQIMpairment:NLDistortion:PM3K <level>  
RFGSignal:CARRier[n]:IQIMpairment:NLDistortion:PM3K?

**Arguments**  
[n]:= <NR1> (“n” determines the carrier index number in the carrier table).  
If omitted, n is interpreted as 1.  
[level]:= <NR3> value.

**Returns**  
A single <NR3> value.

**Examples**  
RFGSIGNAL:CARRIER1:IQIMPAIRMENT:NLDISTORTION:PM3K -1 sets the AM/PM k3 level to –1 dB for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:IQIMPAIRMENT:NLDISTORTION:PM3K? might return -1.0000000000, indicating that the AM/PM k3 level is set to –1 dB for the carrier at index 3.
RFGSignal:CARRier[n]:IQIMpairment:NLDistortion:TON

This command sets or returns the Nonlinear Distortions state (enabled or disabled) for IQ Impairments for the specified carrier in the carrier table.

Group  IQ impairments

Syntax  RFGSignal:CARRier[n]:IQIMpairment:NLDistortion:TON
        {1|0|ON|OFF}
        RFGSignal:CARRier[n]:IQIMpairment:NLDistortion:TON?

Arguments  [n]:= <NR1> (“n” determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

OFF or 0 disables Nonlinear Distortions. OFF or 0 is the default value.

ON or 1 enables Nonlinear Distortions.

Returns  A single <Boolean> value.

Examples  RFGSIGNAL:CARRIER1:IQIMPAIRMENT:NLDISTORTION:TON 1 enables the Nonlinear Distortions for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:IQIMPAIRMENT:NLDISTORTION:TON? might return 0, indicating that the Nonlinear Distortions is disabled for the carrier at index 3.
RFGSignal:CARRier[n]:IQIMpairment:QERRor:ERRor

This command sets or returns the I/Q Error percentage for Quadrature Error for IQ Impairments for the specified carrier in the carrier table.

<table>
<thead>
<tr>
<th>Group</th>
<th>IQ impairments</th>
</tr>
</thead>
</table>
| Syntax    | RFGSignal:CARRier[n]:IQIMpairment:QERRor:ERRor <percentage>  
            | RFGSignal:CARRier[n]:IQIMpairment:QERRor:ERRor?   |
| Arguments | [n]:= <NR1> (“n” determines the carrier index number in the carrier table).  
            | If omitted, n is interpreted as 1.  
            | <percentage>:= <NR3> value.       |
| Returns   | A single <NR3> value. |
| Examples  | RFGSIGNAL:CARRIER1:IQIMPAIRMENT:QERROR:ERROR –11 sets the Quadrature Error I/Q Error percentage to –11 % for the carrier at index 1 in the carrier table.  
            | RFGSIGNAL:CARRIER3:IQIMPAIRMENT:QERROR:ERROR? might return 3.0000000000, indicating that the Quadrature Error I/Q Error percentage is set to 3 % for the carrier at index 3. |
**RFGSignal:CARRier[n]:IQIMpairment:QERRor:TON**

This command sets or returns the Quadrature Error state (enabled or disabled) for IQ Impairments for the specified carrier in the carrier table.

**Group**  
IQ impairments

**Syntax**  
RFGSignal:CARRier[n]:IQIMpairment:QERRor:TON {1|0|ON|OFF}  
RFGSignal:CARRier[n]:IQIMpairment:QERRor:TON?

**Arguments**  
[n]::= <NR1> (“n” determines the carrier index number in the carrier table).  
If omitted, n is interpreted as 1.  
OFF or 0 disables Quadrature Error. OFF or 0 is the default value.  
ON or 1 enables Quadrature Error.

**Returns**  
A single <Boolean> value.

**Examples**  
RFGSIGNAL:CARRIER1:IQIMPAIRMENT:QERROR:TON 1 enables the Quadrature Error for the carrier at index 1 in the carrier table.  
RFGSIGNAL:CARRIER1:IQIMPAIRMENT:QERROR:TON? might return 0, indicating that the Quadrature Error is disabled for the carrier at index 3.
RFGSignal:CARRier[n]:MULTipath:ADDPath (No Query Form)

This command adds the specified number of paths in the Multipath table for the specified carrier in the carrier table.

Group      Multipath

Syntax      RFGSignal:CARRier[n]:MULTipath:ADDPath <paths>

Related Commands
  RFGSignal:CARRier[n]:MULTipath:DELay
  RFGSignal:CARRier[n]:MULTipath:AMPLitude
  RFGSignal:CARRier[n]:MULTipath:PHASE

Arguments
  [n]::= <NR1> (“n” determines the carrier index number in the carrier table).

  If omitted, n is interpreted as 1.

  <paths>::= <NR1> value.

Examples
  RFGSIGNAL:CARRIER1:MULTIPATH:ADDPATH 2 adds 2 new paths at the end of
  the Multipath table for the carrier at index 1 in the carrier table.
RFGSignal:CARRIER[n]:MULTipath:AMPLitude

This command sets or returns the Multipath Amplitude of the currently selected path for the specified carrier in the carrier table.

Group Multipath

Syntax
RFGSignal:CARRIER[n]:MULTipath:AMPLitude <amplitude>
RFGSignal:CARRIER[n]:MULTipath:AMPLitude?

Related Commands RFGSignal:CARRIER[n]:MULTipath:SELPath

Arguments [n]::= <NR1> (“n” determines the carrier index number in the carrier table).
If omitted, n is interpreted as 1.
<amplitude>::= <NR3> value.

Returns A single <NR3> value.

Examples
RFGSIGNAL:CARRIER1:MULTIPATH:AMPLITUDE –10 sets the Multipath Amplitude of the currently selected path to –10 dB for the carrier at index 1 in the carrier table.
RFGSIGNAL:CARRIER3:MULTIPATH:AMPLITUDE? might return –200.0000000000E-3, indicating that the Amplitude of the currently selected path is –0.2 dB for the carrier at index 3.
RFGSignal:CARRier[n]:MULTipath:DElay

This command sets or returns the Multipath Delay in symbols of the currently selected path for the specified carrier in the carrier table.

Group
Multipath

Syntax
RFGSignal:CARRier[n]:MULTipath:DElay <delay>
RFGSignal:CARRier[n]:MULTipath:DElay?

Related Commands
RFGSignal:CARRier[n]:MULTipath:SELPath

Arguments
[n]::= <NR1> (“n” determines the carrier index number in the carrier table).
If omitted, n is interpreted as 1.
<delay>::= <NR3> value.

Returns
A single <NR3> value.

Examples
RFGSignal:CARRIER1:MULTIPATH:DELAY 25.5 sets the Multipath symbol Delay of the currently selected path to 25.5 symbols for the carrier at index 1 in the carrier table.
RFGSignal:CARRIER3:MULTIPATH:DELAY? might return 200.0000000000E+3, indicating that the Multipath Delay of the currently selected path is 200 k symbols for the carrier at index 3.
RFGSignal:CARRier[n]:MULTipath:DELPath (No Query Form)

This command deletes a single path in the Multipath table for the specified carrier in the carrier table.

If the path doesn’t exist, an error is returned.

Group    Multipath

Syntax    RFGSignal:CARRier[n]:MULTipath:DELPath <path>

Arguments  [n]:= <NR1> (“n” determines the carrier index number in the carrier table).

          If omitted, n is interpreted as 1.

          <path>:= <NR1> value.

Examples   RFGSIGNAL:CARRIER1:MULTIPATH:DELPATH 1 deletes path 1 in the Multipath table for the carrier at index 1 in the carrier table.
**RFGSignal:CARRier[n]:MULTipath:PHASE**

This command sets or returns the Multipath Phase of the currently selected path for the specified carrier in the carrier table.

**Group**  
Multipath

**Syntax**  
RFGSignal:CARRier[n]:MULTipath:PHASE <phase>  
RFGSignal:CARRier[n]:MULTipath:PHASE?

**Related Commands**  
RFGSignal:CARRier[n]:MULTipath:SELPath

**Arguments**  
[n]::= <NR1> (“n” determines the carrier index number in the carrier table).  
If omitted, n is interpreted as 1.  
<phase>::= <NR2> value.

**Returns**  
A single <NR2> value.

**Examples**  
RFGSIGNAL:CARRIER1:MULTIPATH:PHASE 10 sets the Multipath Phase of the currently selected path to 10 degrees for the carrier at index 1 in the carrier table.  
RFGSIGNAL:CARRIER3:MULTIPATH:PHASE? might return –20.0000000000, indicating that the Multipath Phase of the currently selected path is set to –20 degrees for the carrier at index 3.


**RFGSignal:CARRier[n]:MULTipath:SELPath**

This command sets or returns the selected MultiPath table index (row) for the specified carrier in the carrier table.

**Group**  
Multipath

**Syntax**  
RFGSignal:CARRier[n]:MULTipath:SELPath <index>  
RFGSignal:CARRier[n]:MULTipath:SELPath?

**Arguments**  
[n]:= <NR1> (“n” determines the carrier index number in the carrier table).  
If omitted, n is interpreted as 1.

<index>::= <NR1> value.

**Returns**  
A single <NR1> value.

**Examples**  
RFGSIGNAL:CARRIER1:MULTIPATH:SELPATH 1 selects the first path in the MultiPath table for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER2:MULTIPATH:SELPATH? might return 2.0000000000, indicating that the second path in the MultiPath table is selected for the carrier at index 3.
RFGSignal:CARRIER[n]:MULTipath:TURNon

This command sets or returns the Multipath state (enabled or disabled) for the specified carrier in the carrier table.

Group  Multipath

Syntax  RFGSignal:CARRIER[n]:MULTipath:TURNon \{1|0|ON|OFF\}
        RFGSignal:CARRIER[n]:MULTipath:TURNon?

Arguments  [n]::= <NR1> ("n" determines the carrier index number in the carrier table).
            If omitted, n is interpreted as 1.
            OFF or 0 disables Multipath. OFF or 0 is the default value.
            ON or 1 enables Multipath.

Returns  A single <Boolean> value.

Examples  RFGSIGNAL:CARRIER1:MULTIPATH:TURNON 1 enables Multipath for the carrier at index 1 in the carrier table.
           RFGSIGNAL:CARRIER3:MULTIPATH:TURNon? might return 0, indicating that the Multipath is disabled for the carrier at index 3.
RFGSignal:CARRier[n]:NOISe:BANDwidth

This command sets or returns the Bandwidth for the Noise Carrier for the specified carrier in the carrier table when the Carrier Type is set to Noise.

Group  Carrier setup

Syntax  
RFGSignal:CARRier[n]:NOISe:BANDwidth <bandwidth>
RFGSignal:CARRier[n]:NOISe:BANDwidth?

Arguments  
[n]::= <NR1> (“n” determines the carrier index number in the carrier table).
If omitted, n is interpreted as 1.
<bandwidth>::= <NR3> value.

Returns  A single <NR3> value.

Examples  
RFGSIGNAL:CARRIER1:NOISE:BANDWIDTH 2.5E6 sets the Bandwidth to 2.5 MHz for the Noise Carrier for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:NOISE:BANDWIDTH? might return 1.5000000000E+6, indicating that the Bandwidth is set to 1.5 MHz for the carrier at index 3.
RFGSignal:Carrier[n]:PHASE

This command sets or returns the Phase for the specified carrier in the carrier table.

Group  
Carrier setup

Syntax  
RFGSignal:Carrier[n]:PHASE <phase>
RFGSignal:Carrier[n]:PHASE?

Arguments  
[n]::= <NR1> (“n” determines the carrier index number in the carrier table).
If omitted, n is interpreted as 1.
<phase>::=<NR1> value.

Returns  
A single <NR2> value.

Examples  
RFGSIGNAL:CARRIER1:PHASE 90 sets the Phase to 90 degrees for the carrier at index 1 in the carrier table.
RFGSIGNAL:CARRIER3:PHASE? might return -1.5000000000, indicating that the Phase is set to –1.5 degrees for the carrier at index 3.
**RFGSignal:CARRier[n]:PRAMp:DTYPe**

This command sets or returns the Duration Units used for the Power Ramp table for the specified carrier in the carrier table.

**Group**  
Power ramp

**Syntax**  
RFGSignal:CARRier[n]:PRAMp:DTYPe {TIMe|SYMBols}  
RFGSignal:CARRier[n]:PRAMp:DTYPe?

**Arguments**  
[n]:= <NR1> ("n" determines the carrier index number in the carrier table).  
If omitted, n is interpreted as 1.

TIMe sets the levels in the power ramp table in terms of time.  
SYMBols sets the levels in the power ramp table in terms of symbols.

**Returns**  
TIM (Time)  
SYMB (Symbols)

**Examples**  
RFGSIGNAL:CARRIER1:PRAMP:DTYPE TIME sets the Duration Units to Time, causing the Power Ramp table entries to reflect time instead of symbols for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:PRAMP:DTYPE? might return SYMB, indicating that the Duration Units is set to Symbols, causing the Power Ramp table entries to reflect time instead of symbols for the carrier at index 3.
RFGSignal:CARRier[n]:PRAMp:DURation

This command sets or returns the Power Ramp Duration for the specified carrier in the carrier table.

Group  Power ramp

Syntax  RFGSignal:CARRier[n]:PRAMp:DURation <duration>
        RFGSignal:CARRier[n]:PRAMp:DURation

Arguments  [n]::= <NR1> (“n” determines the carrier index number in the carrier table).
           If omitted, n is interpreted as 1.
<duration>::= <NR3> value.

Returns  The duration of the power ramp table.

Examples  RFGSIGNAL:CARRIER1:PRAMP:DURATION 10E-6 sets the Power Ramp Duration to 10 μs for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:PRAMP:DURATION? might return 1.0000000000E-3, indicating that the Power Ramp Duration is set to 1 ms for the carrier at index 3.
RFGSignal:CARRier[n]:PRAMp:FUNCTION

This command sets or returns the Power Ramp Function for the specified carrier in the carrier table.

**Group**  
Power ramp

**Syntax**

RFGSignal:CARRier[n]:PRAMp:FUNCTION {LINear|COSine}  
RFGSignal:CARRier[n]:PRAMp:FUNCTION?

**Arguments**

[n]:= <NR1> (“n” determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

LINnear indicates a linear method of calculating the power ramp.

COSine indicates that the method of interpolation is cosine for calculating the power ramp between symbols.

**Returns**

LIN  
COS

**Examples**

RFG SIGNAL:CARRIER1:PRAMP:FUNCTION LINEAR sets the Power Ramp Function to Linear for the carrier at index 1 in the carrier table.

RFG SIGNAL:CARRIER3:PRAMP:FUNCTION? might return COS, indicating that the Power Ramp Function is set to Cosine for the carrier at index 3.
RFGSignal:Carrier[n]:PRAMp:ILEVel

This command sets or returns the Power Ramp Initial Level of the currently selected hop for the specified carrier in the carrier table.

**Group**  
Power ramp

**Syntax**  
RFGSignal:Carrier[n]:PRAMp:ILEVel <initial_level>  
RFGSignal:Carrier[n]:PRAMp:ILEVel?

**Arguments**  
[n]:= <NR1> ("n" determines the carrier index number in the carrier table).  
If omitted, n is interpreted as 1.  
<initial_level>:= <NR3> value.

**Returns**  
A single <NR3> value.

**Examples**  
RFGSIGNAL:Carrier1:PRAMP:ILEVEL –0.08 sets the Power Ramp Initial Level to –0.08 dB for the carrier at index 1 in the carrier table.  
RFGSignal:CARRier[n]:PRAMp:LEVel:ADD (No Query Form)

This command adds new entries to the Power Ramp table for the specified carrier in the carrier table.

Group  
Power ramp

Syntax  
RFGSignal:CARRier[n]:PRAMp:LEVel:ADD <entries>
RFGSignal:CARRier[n]:PRAMp:LEVel:ADD

Related Commands  
RFGSignal:CARRier[n]:PRAMp:LEVel:DURation
RFGSignal:CARRier[n]:PRAMp:LEVel:ESYMbol
RFGSignal:CARRier[n]:PRAMp:LEVel:POWer
RFGSignal:CARRier[n]:PRAMp:LEVel:SSYMbol

Arguments  
[n]:= <NR1> ("n" determines the carrier index number in the carrier table).
If omitted, n is interpreted as 1.
<entries>:= <NR1> value.
The maximum number of table entries is 21.

Examples  
RFGSIGNAL:CARRIER1:PRAMP:LEVEL:ADD 2 adds two entries to the end of the Power Ramp table for the carrier at index 1 in the carrier table.
RFGSignal:CARRier[n]:PRAMp:LEVel:DELeTe (No Query Form)

This command deletes entries from the Power Ramp table for the specified carrier in the carrier table. An error is returned if the entry does not exist.

Group       Power ramp

Syntax      RFGSignal:CARRier[n]:PRAMp:LEVel:DELeTe {<index>|ALL}

Arguments   [n]::= <NR1> ("n" determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

<index>::= <NR1> value.

ALL deletes all table entries.

Examples    RFGSIGNAL:CARRIER1:PRAMP:LEVEL:DELETE 1 deletes the first entry from
            the Power Ramp table for the carrier at index 1 in the carrier table.
RFGSignal:CARRier[n]:PRAMp:LEVel:DURation

This command sets or returns the Duration of the currently selected Power Ramp table index for the specified carrier in the carrier table.

The Power Ramp Duration Units must be set to Time.

Group  Power ramp

Syntax  
RFGSignal:CARRier[n]:PRAMp:LEVel:DURation <duration>
RFGSignal:CARRier[n]:PRAMp:LEVel:DURation?

Related Commands  
RFGSignal:CARRier[n]:PRAMp:DTYPe
RFGSignal:CARRier[n]:PRAMp:LEVel:SELect

Arguments  
[n]::= <NR1> (“n” determines the carrier index number in the carrier table).
If omitted, n is interpreted as 1.
<duration>::= <NR3> value.

Returns  A single <NR3> value.

Examples  
RFGSIGNAL:CARRIER1:PRAMP:LEVEL:DURATION 1E-6 sets the Duration, of the currently selected Power Ramp table index, to 1 ms for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:PRAMP:LEVEL:DURATION? might return 10.0000000000E-6, indicating that the Duration, of the currently selected Power Ramp table index, is set to 10 μs for the carrier at index 3.
**RFGSignal:CARRier[n]:PRAMp:LEVel:ESYMbol**

This command sets or returns the End Symbol of the currently selected Power Ramp table index for the specified carrier in the carrier table.

The Duration Units must be set to Symbols.

**Group**
Power ramp

**Syntax**
- `RFGSignal:CARRIER[n]:PRAMP:LEVEL:ESYMbol <end_symbol>`
- `RFGSignal:CARRIER[n]:PRAMP:LEVEL:ESYMbol?`

**Related Commands**
- `RFGSignal:CARRIER[n]:PRAMP:DTYPE`
- `RFGSignal:CARRIER[n]:PRAMP:LEVEL:SELECT`

**Arguments**
- `[n]::= <NR1>` ("n" determines the carrier index number in the carrier table).
  - If omitted, n is interpreted as 1.
- `<end_symbol>::= <NR3>` value.

**Returns**
A single `<NR3>` value.

**Examples**
- `RFGSIGNAL:CARRIER1:PRAMP:LEVEL:SYMBOL 25` sets End Symbol, of the currently selected Power Ramp table index, to symbol 25 for the carrier at index 1 in the carrier table.
- `RFGSIGNAL:CARRIER3:PRAMP:LEVEL:SYMBOL?` might return `45.0000000000`, indicating that the End Symbol, of the currently selected Power Ramp table index, is set to symbol 45 for the carrier at index 3.
RFGSignal:CARRier[n]:PRAMp:LEVel:POWer

This command sets or returns the Power Level of the currently selected Power Ramp table index for the specified carrier in the carrier table.

**Group**

Power ramp

**Syntax**

RFGSignal:CARRier[n]:PRAMp:LEVel:POWer <power>

RFGSignal:CARRier[n]:PRAMp:LEVel:POWer?

**Related Commands**

RFGSignal:CARRier[n]:PRAMp:LEVel:SELect

**Arguments**

[n]::= <NR1> (“n” determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

<power>::= <NR3> value.

**Returns**

A single <NR3> value.

**Examples**

RFGSIGNAL:CARRIER1:PRAMP:LEVEL:POWER –4 sets the Power Level, of the currently selected Power Ramp table index, to –4 dB for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:PRAMP:LEVEL:POWER? might return –5.5000000000, indicating that the Power Level, of the currently selected Power Ramp table index, is set to –5.5 dB for the carrier at index 3.
RFGSignal:CARRier[n]:PRAMp:LEVel:SELect

This command sets or returns the selected index (row) of the Power Ramp table for the specified carrier in the carrier table.

Group   Power ramp

Syntax   RFGSignal:CARRier[n]:PRAMp:LEVel:SELect <index>
          RFGSignal:CARRier[n]:PRAMp:LEVel:SELect?

Arguments [n]:= <NR1> (“n” determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

<index>: = <NR1> value.

Returns   A single <NR1> value.

Examples   RFGSIGNAL:CARRIER1:PRAMP:LEVEL:SELECT 2 selects the second entry of
          the Power Ramp table for the carrier at index 1 in the carrier table.

          RFGSIGNAL:CARRIER3:PRAMP:LEVEL:SELECT? might return 3, indicating that
          the third entry of the Power Ramp is selected table for the carrier at index 3.
**RFGSignal:CARRier[n]:PRAMp:LEVel:SSYMbol**

This command sets or returns the Start Symbol of the currently selected Power Ramp table index for the specified carrier in the carrier table.

The Duration Units must be set to Symbols.

**Group**  
Power ramp

**Syntax**

RFGSignal:CARRier[n]:PRAMp:LEVel:SSYMbol <start_symbol>
RFGSignal:CARRier[n]:PRAMp:LEVel:SSYMbol?

**Related Commands**

RFGSignal:CARRier[n]:PRAMp:DTYPe  
RFGSignal:CARRier[n]:PRAMp:LEVel:SELect

**Arguments**

[n]::= <NR1> (“n” determines the carrier index number in the carrier table).

If omitted, n is interpreted as 1.

<start_symbol>::= <NR3> value.

**Returns**

A single <NR3> value.

**Examples**

RFGSIGNAL:CARRIER1:PRAMP:LEVEL:SSYMBOL 3 sets the Start Symbol, of the currently selected Power Ramp table index, to symbol 3 for the carrier at index 1 in the carrier table.

RFGSIGNAL:CARRIER3:PRAMP:LEVEL:SSYMBOL? might return 15.0000000000, indicating that the Start Symbol, of the currently selected Power Ramp table index, is set to symbol 15 for the carrier at index 3.
**RFGSignal:CARRier[n]:PRAMp:PEXTend**

This command sets or returns the Periodic Extend Power Levels state (enabled or disabled) for the Power Ramp table for the specified carrier in the carrier table.

**Group**  
Power ramp

**Syntax**  
RFGSignal:CARRier[n]:PRAMp:PEXTend {1|0|ON|OFF}  
RFGSignal:CARRier[n]:PRAMp:PEXTend?

**Arguments**  
[n]:= <NR1> (“n” determines the carrier index number in the carrier table).  
If omitted, n is interpreted as 1.  
OFF or 0 disables the Periodic Extend Power Levels setting. OFF or 0 is the default value.  
ON or 1 enables the Periodic Extend Power Levels setting.

**Returns**  
A single <Boolean> value.

**Examples**  
RFGSIGNAL:CARRIER1:PRAMP:PEXTEND 1 enables the Periodic Extend Power Levels of the Power Ramp table for the carrier at index 1 in the carrier table.  
RFGSIGNAL:CARRIER3:PRAMP:PEXTEND? might return 0, indicating that the Periodic Extend Power Levels of the Power Ramp table is disabled for the carrier at index 3.
RFGSignal:CARRier[n]:PRAMp:TON

This command sets or returns the Power Ramp state (enabled or disabled) for the specified carrier in the carrier table.

Group
Power ramp

Syntax
RFGSignal:CARRier[n]:PRAMp:TON {1|0|ON|OFF}
RFGSignal:CARRier[n]:PRAMp:TON?

Arguments
[n]:= <NR1> (“n” determines the carrier index number in the carrier table).
If omitted, n is interpreted as 1.
OFF or 0 disables Power Ramp. OFF or 0 is the default value.
ON or 1 enables Power Ramp.

Returns
A single <Boolean> value.

Examples
RFGSIGNAL:CARRIER1:PRAMP:TON ON enables Power Ramp for the carrier at index 1 in the carrier table.
RFGSIGNAL:CARRIER3:PRAMP:TON? might return 0, indicating that Power Ramp is disabled for the carrier at index 3.
**RFGSignal:CARRier[n]:RFAMplitude**

This command sets or returns the Amplitude for the specified carrier in the carrier table.

**Group**  
Carrier setup

**Syntax**  
RFGSignal:CARRier[n]:RFAMplitude <amplitude>  
RFGSignal:CARRier[n]:RFAMplitude?

**Arguments**  
[n]:= <NR1> (“n” determines the carrier index number in the carrier table).  
If omitted, n is interpreted as 1.  
<amplitude>::=<NR2> value.

**Returns**  
A single <NR2> value.

**Examples**  
RFGSignal:CARRier1:RFAMplitude -3 sets the Amplitude to -3 dBm for the carrier at index 1 in the carrier table.  
RFGSignal:CARRier2:RFAMplitude? might return -12.0000000000, indicating that the Amplitude is set to –12 dBm. for the carrier at index 2
**RFGSignal:CARRier\([n]\):TYPE**

This command sets or returns the Carrier Type for the specified carrier in the carrier table.

**Group**  
Carrier setup

**Syntax**  
```
RFGSignal:CARRier\([n]\):TYPE
{DMODulation|AMODulation|NOISe|CMODulation}
RFGSignal:CARRier\([n]\):TYPE?
```

**Arguments**  
\([n]\)::= <NR1> (“n” determines the carrier index number in the carrier table).

- If omitted, \(n\) is interpreted as 1.
- DMODulation = Digital Modulation.
- AMODulation = Analog Modulation.
- NOISe = Noise.
- CMODulation = Custom Modulation.

**Returns**  
DMOD  
AMOD  
NOIS  
CMOD

**Examples**  
```
RFGSIGNAL:CARRIER1:TYPE DMOD sets the Carrier Type to Digital Modulation for the carrier at index 1 in the carrier table.
RFGSIGNAL:CARRIER3:TYPE? might return DMOD, indicating that the Carrier Type is set to Digital Modulation for the carrier at index 3.
```
RFGSignal:COMPile (No Query Form)

This command compiles and generates a waveform using the RF Generic Signal plug-in compile settings.

**Conditions**
The active plug-in must be RF Generic Signal.

This is an overlapping command. Overlapping commands run concurrently with other commands, allowing additional commands to start before the overlapping command has finished.

**Group**
Compile

**Syntax**
RFGSignal:COMPile

**Examples**
RFGSIGNAL:COMPILE compiles and generates the waveforms.
**RFGSignal:COMPlie:AWARound**

This command sets or returns the Adjust Wrap Around state (enabled or disabled) for the Compile Settings.

**Group**  
Compile

**Syntax**  
RFGSignal:COMPlie:AWARound {0|1|OFF|ON}  
RFGSignal:COMPlie:AWARound?

**Arguments**  
OFF or 0 disables Adjust Wrap Around.  
ON or 1 enables Adjust Wrap Around. ON or 1 is the default value.

**Returns**  
A single <Boolean> value.

**Examples**  
RFGSIGNAL:COMPILE:AWAROUND ON enables Adjust Wrap Around.  
RFGSIGNAL:COMPILE:AWAROUND? might return 0, indicating Adjust Wrap Around is disabled.
**RFGSignal:COMPile:CANCel (No Query Form)**

This command cancels a compilation currently in progress.

**Conditions**

The active plug-in must be RF Generic Signal.

**Group**

Compile

**Syntax**

RFGSignal:COMPile:CANCel

**Examples**

RFGSIGNAL:COMPILE:CANCEL immediately ends the current compile process.
RFGSignal:COM:PILE:CASSign

This command sets or returns the state (enabled or disabled) to compile the waveform and immediately assign it to a specified channel (enabled) or just compile the waveform (disabled).

**Group**  Compile

**Syntax**

```
RFGSignal:COM:PILE:CASSign {0|1|OFF|ON}
RFGSignal:COM:PILE:CASSign?
```

**Related Commands**

- RFGSignal:COM:PILE:CHAN:RF
- RFGSignal:COM:PILE:CHAN:I
- RFGSignal:COM:PILE:CHAN:Q

**Arguments**

0 or OFF will only compile the waveform.
1 or ON will compile and assign the waveform to a channel.

**Returns**

A single <Boolean> value.

**Examples**

```
RFGSIGNAL:COMPILE:CASSIGN 1 enables the compile and assign function.
RFGSIGNAL:COMPILE:CASSIGN? might return 0, indicating that the compile and assign function is disabled.
```
RFGSignal:COMPile:CHANnel:I

This command sets or returns the playout channel intended for the compiled I signal waveform when the signal format is set to IQ. The selected channel is also used to define the amplitude ranges.

If RFGSignal:COMPile:CASSign is enabled, the waveform is assigned to the specified channel.

Group  Compile

Syntax  
RFGSignal:COMPile:CHANnel:I <channel>
RFGSignal:COMPile:CHANnel:I?

Related Commands  
RFGSignal:COMPile:CHANnel:Q,
RFGSignal:COMPile:CASSign,
RFGSignal:COMPile:PLAY

Arguments  
<channel>::=<NR1> is a valid channel number.

Returns  
A single <NR1> value.

Examples  
RFGSIGNAL:COMPILE:CHANNEL:I 2 assigns the I signals to channel 2.
RFGSIGNAL:COMPILE:CHANNEL:I? might return 1, indicating that I signals are assigned to channel 1.
**RFGSignal:COMPile:CHANnel:IQ**

This command sets or returns the playout channel intended for the compiled IQ signal after up-conversion using the internal IQ modulator.

The selected channel is also used to define the amplitude ranges.

**Conditions**
The Signal Format must be set to IQ and the instrument must have an internal IQ modulator and be enabled.

**Group**
Compile

**Syntax**

```
RFGSignal:COMPile:CHANnel:IQ <channel>
RFGSignal:COMPile:CHANnel:IQ?
```

**Related Commands**
```
RFGSignal:COMPile:DUPConverter
```

**Arguments**

<channel>::= <NR1> is a valid channel number.

Channel 1 is the default channel.

**Returns**

A single <NR1> value.

**Examples**

```
RFGSIGNAL:COMPILE:CHANNEL:IQ 2 compiles the IQ signals for channel 2.
```

```
RFGSIGNAL:COMPILE:CHANNEL:IQ? might return 1, indicating that IQ signals are compiled for channel 1.
```
RFGSignal:COMPile:CHANnel:Q

This command sets or returns the playout channel intended for the compiled Q signal waveform when the signal format is set to IQ. The selected channel is also used to define the amplitude ranges.

If RFGSignal:COMPile:CASSign is enabled, the waveform is assigned to the specified channel.

Group    Compile

Syntax

RFGSignal:COMPile:CHANnel:Q <channel>
RFGSignal:COMPile:CHANnel:Q?

Related Commands

RFGSignal:COMPile:CHANnel:I,
RFGSignal:COMPile:CASSign,
RFGSignal:COMPile:PLAY

Arguments

<channel>::=<NR1> is a valid channel number.

Returns

A single <NR1> value.

Examples

RFGSIGNAL:COMPILE:CHANNEL:Q 2 assigns the Q signals to channel 2.
RFGSIGNAL:COMPILE:CHANNEL:Q? might return 1, indicating that Q signals are assigned to channel 1.
RFGSignal:COMPile:CHANnel:RF

This command sets or returns the playout channel intended for the compiled RF signal waveform when the signal format is set to RF/IF. The selected channel is also used to define the amplitude ranges.

If RFGSignal:COMPile:CASSign is enabled, the waveform is assigned to the specified channel.

Group       Compile

Syntax      RFGSignal:COMPile:CHANnel:RF <channel>
            RFGSignal:COMPile:CHANnel:RF?

Related Commands    RFGSignal:COMPile:CASSign,
                    RFGSignal:COMPile:PLAY

Arguments    <channel>::=<NR1> is a valid channel number.

Returns      A single <NR1> value.

Examples     RFGSIGNAL:COMPILE:CHANNEL:RF 2 assigns the RF signals to channel 2.
             RFGSIGNAL:COMPILE:CHANNEL:RF? might return 1, indicating that RF signals are assigned to channel 1.
**RFGSignal:COMPile:CORRection:APPLY**

This command sets or returns the Apply Corrections File state (enabled or disabled) for the Compile Settings.

When applying correction files, you have the capability adjust the frequency response.

- When applying an RF correction file, you can apply a Gaussian filter/bandwidth or remove Sin(x)/x distortions.
- When applying an I/Q correction file, you can apply a skew.

These settings are global settings controlled by the host instrument. The PI commands to access these settings are provided in the host instrument’s programmer manual.

**Group**

Compile

**Syntax**

```
RFGSignal:COMPile:CORRection:APPLY \{1|0|ON|OFF\}
RFGSignal:COMPile:CORRection:APPLY?
```

**Related Commands**

RFGSignal:COMPile:CORRection:PATH

**Arguments**

OFF or 0 disables Apply Corrections File. OFF or 0 is the default value.

ON or 1 enables Apply Corrections File.

**Returns**

A single <Boolean> value.

**Examples**

```
RFGSIGNAL:COMPILE:CORRECTION:APPLY ON enables Apply Corrections File.

RFGSIGNAL:COMPILE:CORRECTION:APPLY? might return 0, indicating that Apply Corrections File is disabled.
```
RFGSignal:COMPILE:CORRECTION:PATH

This command sets or returns the RF Correction filename and filepath to use when compiling an RF signal.

**Conditions**

The signal format must be set to RF.
The Precompensation plug-in is required.

**Group**

Compile

**Syntax**

RFGSignal:COMPILE:CORRECTION:PATH <filepath>

RFGSignal:COMPILE:CORRECTION:PATH?

**Arguments**

filepath::=<string> defines path to the corrections file.

**Returns**

A single <filepath> string.

**Examples**

RFGSIGNAL:COMPILE:CORRECTION:PATH
"C:\temp\CorrectionFile.corr" sets the RF Correction filename and filepath.

RFGSIGNAL:COMPILE:CORRECTION:PATH? might return
"C:\temp\CorrectionFile.corr" as the RF Correction filename and filepath.
RFGSignal:COMPile:CORRection:PATH:I

This command sets or returns the I Correction filename and filepath to use when compiling and IQ signal.

Conditions
The signal format must be set to IQ.
Apply corrections must be enabled.

Group
Compile

Syntax
RFGSignal:COMPile:CORRection:PATH:I <filepath>

Related Commands
RFGSignal:COMPile:CORRection:TYPE,
RFGSignal:COMPile:CORRection:APPLy

Arguments
<filepath>::=<string> defines path to the I correction file.

Returns
A single <filepath> string.

Examples
RFGSIGNAL:COMPILE:CORRECTION:PATH:I
"C:\temp\ICorrectionFile.corr" sets the I Correction filename and filepath.

RFGSIGNAL:COMPILE:CORRECTION:PATH:I?
"C:\temp\ICorrectionFile.corr" as the I Correction filename and filepath.
RFGSignal:COMPile:CORRection:PATH:IQ

This command sets or returns the IQ Correction filename and filepath to use when compiling and IQ signal.

Conditions
- The signal format must be set to IQ.
- Apply corrections must be enabled.

Group
- Compile

Syntax
RFGSignal:COMPile:CORRection:PATH:IQ <filepath>

Related Commands
- RFGSignal:COMPile:CORRection:TYPE,
- RFGSignal:COMPile:CORRection:APPLY

Arguments
<filepath>::=<string> defines path to the IQ correction file.

Returns
A single <filepath> string.

Examples
RFGSIGNAL:COMPILE:CORRECTION:PATH:IQ
"C:\temp\ICorrectionFile.corr" sets the IQ Correction filename and filepath.

RFGSIGNAL:COMPILE:CORRECTION:PATH:IQ? might return
"C:\temp\ICorrectionFile.corr" as the IQ Correction filename and filepath.
**RFGSignal:COMPile:CORRection:PATH:Q**

This command sets or returns the Q correction filename and filepath to use when compiling and IQ signal.

**Conditions**
The signal format must be set to IQ.
Apply corrections must be enabled.

**Group**
Compile

**Syntax**
RFGSignal:COMPile:CORRection:PATH:Q <filepath>

**Related Commands**
RFGSignal:COMPile:CORRection:TYPE,
RFGSignal:COMPile:CORRection:APPLy

**Arguments**
<filepath>::=<string> defines path to the Q correction file.

**Returns**
A single <filepath> string.

**Examples**
RFGSIGNAL:COMPILE:CORRECTION:PATH:Q
"C:\temp\QCorrectionFile.corr" sets the Q Correction filename and filepath.

"C:\temp\QCorrectionFile.corr" as the Q Correction filename and filepath.
RFGSignal:COMPile:CORRection:TYPE

This command sets or returns the type of IQ correction file (a single IQ file or individual I and Q files) to apply when compiling.

Conditions  The signal format must be set to IQ.

Group  Compile

Syntax  

\[
\text{RFGSignal:COMPile:CORRection:TYPE } \{\text{IQ|BOTH}\} \\
\text{RFGSignal:COMPile:CORRection:TYPE?}
\]

Arguments  IQ: The compile process uses a single IQ correction file.

BOTH: The compile process uses both an I correction file and a Q correction file.

Returns  IQ|BOTH

Examples  \[\text{RFGSIGNAL:COMPILE:CORRECTION:TYPE } \text{IQ}\] sets the compiler to apply a single IQ correction file when compiling.

\[\text{RFGSIGNAL:COMPILE:CORRECTION:TYPE?}\] might return BOTH, indicating that the compiler will apply both an I and Q correction file to the I and Q signals.
RFGSignal:COMPile:DUPConverter

This command sets or returns the Internal IQ Modulator state (enabled or disabled) when compiling.

When enabled, a complex IQ Waveform is created which can be used with the internal IQ modulator. Sampling Rate and interpolation rates will be calculated based on the Baseband parameters.

When disabled, a separate I and Q signals are created.

**Conditions**
The instrument must have an internal IQ Modulator and the Signal Format set to IQ.

**Group**
Compile

**Syntax**
```
RFGSignal:COMPile:DUPConverter {0|1|OFF|ON}
RFGSignal:COMPile:DUPConverter?
```

**Arguments**
OFF or 0 disables the internal IQ Modulator. OFF or 0 is the default value.
ON or 1 enables the internal IQ Modulator.

**Returns**
A single <Boolean> value.

**Examples**
```
RFGSIGNAL:COMPILE:DUPCONVERTER ON enables the internal IQ Modulator.
RFGSIGNAL:COMPILE:DUPCONVERTER? might return 0, indicating the internal IQ Modulator is disabled.
```
RFGSignal:COMPile:FDRange

This command sets or returns the Fit to full dynamic range state (enabled or disabled) for the Compile Settings.

**Group**  
Compile

**Syntax**  
RFGSignal:COMPile:FDRange {1|0|ON|OFF}  
RFGSignal:COMPile:FDRange?

**Arguments**  
OFF or 0 disables Fit to full dynamic range.  
ON or 1 enables Fit to full dynamic range. ON or 1 is the default value.

**Returns**  
A single <Boolean> value.

**Examples**  
RFGSIGNAL:COMPILE:FDRANGE ON enables Fit to dynamic range.  
RFGSIGNAL:COMPILE:FDRANGE? might 0, indicating Fit to dynamic range is disabled.
RFGSignal:COMPile:ICHannel

NOTE. This command exists for backwards compatibility. Use the command RFGSignal:COMPile:CHANnel:I.

This command sets or returns which channel the I signal is assigned to upon compile when the signal format is set to IQ.

**Group**
Compile

**Syntax**
RFGSignal:COMPile:ICHannel {NONE|<channel>}
RFGSignal:COMPile:ICHannel?

**Arguments**
NONE indicates that the I signal will be compiled but not assigned to a channel.

<channel>::=<NR1> is a valid channel number. The I signal is compiled and assigned to the channel.

**Returns**
A single <NR1> value or NONE.

**Examples**
RFGSIGNAL:COMPILE:ICHANNEL 2 assigns the I signals to channel 2 after the compile completes.

RFGSIGNAL:COMPILE:ICHANNEL? might return NONE, indicating that I signals are not assigned to any channel after the compile is complete.
**RFGSignal:COMPILE:MARKer:ENABLE**

This command sets or returns the Marker Data state (enabled or disabled) for the Compile Settings.

**Group**  
Compile

**Syntax**  
RFGSignal:COMPILE:MARKer:ENABLE \{1|0|ON|OFF\}  
RFGSignal:COMPILE:MARKer:ENABLE?

**Related Commands**  
RFGSignal:COMPILE:MARKer[n]:MDATa  
RFGSignal:COMPILE:MARKer[n]:CFrequency

**Arguments**  
OFF or 0 disables Marker Data. OFF or 0 is the default value.  
ON or 1 enables Marker Data.

**Returns**  
A single <Boolean> value.

**Examples**  
RFGSIGNAL:COMPILE:MARKER:ENABLE ON enables the Marker Data.  
RFGSIGNAL:COMPILE:MARKER:ENABLE? might return 0, indicating that Marker Data is disabled.
**RFGSignal:COMPile:MARKer[n]:CFRequency**

This command sets or returns the Marker Data clock frequency when the Marker Data type is set to Clock Frequency.

**Group**  
Compile

**Syntax**  
RFGSignal:COMPile:MARKer[n]:CFRequency <frequency>  
RFGSignal:COMPile:MARKer[n]:CFRequency?

**Arguments**  
<frequency>::=<NR3> value.

**Returns**  
A single <NR3> value.

**Examples**  
RFGSIGNAL:COMPILE:MARKER1:CFREQUENCY 1E9 sets the Marker 2 Data clock frequency to 1 GHz.  
RFGSIGNAL:COMPILE:MARKER2:CFREQUENCY? might return 10.0000000000E+6, indicating the Marker 2 clock frequency is set to 10 MHz.
RFGSignal:COMPile:MARKer[n]:MDATa

This command sets or returns the Marker Data Type of the specified marker.

Group
Compile

Syntax
RFGSignal:COMPile:MARKer[n]:MDATa {SRATe|BRATe|CFRequency}
RFGSignal:COMPile:MARKer[n]:MDATa?

Related Commands
RFGSignal:COMPile:MARKer:ENABle

Arguments
[n] ::= specifies the Marker. The number of markers available for the channel is dependent on the instrument model.

If omitted, n is interpreted as 1.
SRATe sets the marker data type sample rate.
BRATe sets the marker data type bit rate.
CFRequency sets the marker data type to clock frequency.

Returns
SRAT
BRAT
CFR

Examples
RFGSIGNAL:COMPILE:MARKER1:MDATA BRATE sets the Marker 1 Data type to Bit Rate.
RFGSIGNAL:COMPILE:MARKER2:MDATA? might return CFR, indicating that the Marker 2 Data type is based on the clock frequency provided.
**RFGSignal:COMPile:NAMe**

This command sets or returns the name of the compiled waveform.

**Group**  
Compile

**Syntax**  
RFGSignal:COMPile:NAMe <signal_name>  
RFGSignal:COMPile:NAMe?

**Arguments**  
<signal_name>::=<string> defines the signal name to be created.

**Returns**  
<string>

**Examples**  
RFGSIGNAL:COMPILE:NAME "RFGenWfm" sets the waveform name to RFGenWfm.  
RFGSIGNAL:COMPILE:NAME? might return "RFGenWfm".
**RFGSignal:COMPile:OSAMplng**

This command sets or returns the over sampling rate used to determine the sampling rate of the compiled signal.

The Sampling Rate is calculated by multiplying the Over sampling value with the maximum frequency of the signal to be generated.

**Group**  
Compile

**Syntax**  
RFGSignal:COMPile:OSAMplng <oversampling>  
RFGSignal:COMPile:OSAMplng?

**Arguments**  
Oversampling>::=<NR1> value.  
Range: 1 to 1000

**Returns**  
A single <NR1> value.

**Examples**  
RFGSIGNAL:COMPILE:OSAMPLING 6 sets the over sampling to 6.  
RFGSIGNAL:COMPILE:OSAMPLING? might return 4, indicating that the over sampling rate is set to 4.

**RFGSignal:COMPile:PLAY**

This command sets or returns the Play after assign state (enabled or disabled) for the Compile Settings.

Play after assign is active only when Compile and assign is enabled.

**Group**  
Compile

**Syntax**  
RFGSignal:COMPile:PLAY {0|1|OFF|ON}  
RFGSignal:COMPile:PLAY?

**Arguments**  
OFF or 0 disables Play after assign. OFF or 0 is the default value.  
ON or 1 enables Play after assign.

**Returns**  
A single <Boolean> value.
Examples

RFGSIGNAL:COMPILE:PLAY 1 enables Play after assign.

RFGSIGNAL:COMPILE:PLAY? might return 0, indicating that Play after assign is disabled.
**RFGSignal:COMPile:QChannel**

**NOTE.** This command exists for backwards compatibility. Use the command `RFGSignal:COMPile:CHANnel:Q`.

This command sets or returns which channel the Q signal is assigned to upon compile when the signal format is set to IQ.

**Group**  Compile

**Syntax**

```
RFGSignal:COMPile:QChannel {NONE|<channel>}
RFGSignal:COMPile:QChannel?
```

**Arguments**

NONE indicates that the Q signal will be compiled but not assigned to a channel.

<channel>::= <NR1> is a valid channel number. The Q signal is compiled and assigned to the channel.

**Returns**

A single <NR1> value or NONE.

**Examples**

```
RFGSIGNAL:COMPILE:QCHANNEL 2 assigns the Q signals to channel 2 after the compile completes.
```

```
RFGSIGNAL:COMPILE:QCHANNEL? might return NONE, indicating that Q signals are not assigned to any channel after the compile is complete.
```
RFGSignal:COMpIle:RFCHannel

NOTE. This command exists for backwards compatibility. Use the command RFGSignal:COMpIle:CHANnel:RF.

This command sets or returns which channel the RF waveform is assigned to upon compile when the signal format is set to RF.

Group   Compile

Syntax  RFGSignal:COMpIle:RFCHannel {NONE|<channel>}
        RFGSignal:COMpIle:RFCHannel?

Arguments  NONE indicates that the waveform will be compiled but not assigned to a channel.
            <channel>::=<NR1> is a valid channel number. The waveform is compiled and assigned to the channel.

Returns  A single <NR1> value or NONE.

Examples  RFGSIGNAL:COMPILE:RFCHANNEL 2 assigns RF waveforms to channel 2 after the compile completes.

          RFGSIGNAL:COMPILE:RFCHANNEL? might return NONE, indicating that RF waveforms are not assigned to any channel after the compile is complete.
**RFGSignal:COMPile:SFORmat**

This command sets or returns the signal format of the signal to be compiled.

**Group**  
Compile

**Syntax**  
```
RFGSignal:COMPile:SFORmat {RF|IQ}
RFGSignal:COMPile:SFORmat?
```

**Arguments**  
- **RF** sets the signal format to RF.
- **IQ** sets the signal format to IQ.

**Returns**  
- **RF**
- **IQ**

**Examples**  
- `RFGSIGNAL:COMPILE:SFORMAT IQ` sets the signal format to IQ mode.
- `RFGSIGNAL:COMPILE:SFORMAT?` might return **RF**, indicating that the signal format is set to RF mode.
RFGSignal:COMPile:SRATe

This command sets or returns the sampling rate for the compile settings.

**Group**
- Compile

**Syntax**
- RFGSignal:COMPile:SRATe <rate>
- RFGSignal:COMPile:SRATe?

**Arguments**
- <rate>::= <NR3> value.
  - The maximum sampling rate is dependent on the instrument and instrument options.

**Returns**
- A single <NR3> value.

**Examples**
- RFGSIGNAL:COMPILE:SRATE 6E9 sets the Sampling Rate to 6 GS/s in the Compile settings.
- RFGSIGNAL:COMPILE:SRATE? might return 6.0000000000E+9, indicating that the sampling rate is GS/s
**RFGSignal:COMPile:SRATe:AUTO**

This command sets or returns the value that indicates if the sampling rate will be automatically calculated at compile time.

**Group**  
Compile

**Syntax**  
RFGSignal:COMPile:SRATe:AUTO {0|1|OFF|ON}  
RFGSignal:COMPile:SRATe:AUTO?

**Arguments**  
OFF or 0 disables the sampling rate auto calculation and the manual setting for is used.  
ON or 1 enables the auto calculation. ON or 1 is the default value.

**Returns**  
A single <Boolean> value.

**Examples**  
RFGSIGNAL:COMPILE:SRATE:AUTO ON sets the signal's sampling rate to be automatically calculated at compile time.  
RFGSIGNAL:COMPILE:SRATE:AUTO? might return 0, indicating that the sampling rate will not be automatically calculated at compile time.
RFGSignal:COMPile:WLENgth

This command sets or returns the waveform length. If the waveform length is not set to auto then the value for waveform length is based on the value provided.

Group   Compile

Syntax   RFGSignal:COMPile:WLENgth <length>
         RFGSignal:COMPile:WLENgth?

Related Commands   RFGSignal:COMPile:WLENgth:TYPE

Arguments   <length>::= <NR3> value.

Returns   A single <NR3> value.

Examples   RFGSIGNAL:COMPILE:WLENGTH 10E-3 sets the waveform length to 10000 samples if the unit is set to samples.

         RFGSIGNAL:COMPILE:WLENGTH? might return 10.0000000000E+3, and if the unit is time, it indicates that the waveform length is 10000 samples.
**RFGSignal:COMPlie:WLENgth:AUTO**

This command sets or returns if the waveform length will be automatically calculated at compile time.

**Group**  
Compile

**Syntax**

RFGSignal:COMPlie:WLENgth:AUTO {0|1|OFF|ON}  
RFGSignal:COMPlie:WLENgth:AUTO?

**Arguments**  
OFF or 0 disables waveform length auto calculation and manual setting for .  
ON or 1 enables the auto calculation. ON or 1 is the default value.

**Returns**  
A single <Boolean> value.

**Examples**

RFGSIGNAL:COMPILE:WLENGTH:AUTO ON sets the signal's waveform length to be automatically calculated at compile time.

RFGSIGNAL:COMPILE:WLENGTH:AUTO? might return 0, indicating that the waveform length will not be automatically calculated at compile time.
**RFGSignal:COMPile:WLENgth:TYPE**

This command sets or returns the waveform length unit type that is used to set the waveform length. If the waveform length is not set to auto then the value for waveform length is based on the value provided. The waveform length can be provided in terms of samples, symbols, or seconds.

**Group**
Compile

**Syntax**

```
RFGsignal:COMPile:WLENgth:TYPE {SAMPles|SYMBols|TIME}
RFGsignal:COMPile:WLENgth:TYPE?
```

**Arguments**

SAMPles sets the waveform length unit type to be samples.
TIME sets the waveform length unit type to be seconds.
SYMBols sets the waveform length unit type to be symbols

**Returns**
SAMP
SYMB
TIME

**Examples**

```
RFGSIGNAL:COMPILE:WLENGTH:TYPE SAMPles sets the waveform length unit type to be samples.
RFGSIGNAL:COMPILE:WLENGTH:TYPE? might return TIME indicating that the waveform length unit is in terms of seconds.
```
**RFGSignal:COMPlie:WOVerwrite**

This command sets or returns the state (enabled or disabled) whether or not to overwrite the existing waveform if the waveform already exists in the Waveform list.

When disabled, a new compiled waveform using the same name will have a numeric suffix added to the end of the waveform name.

**Group**  
Compile

**Syntax**  
```
RFGSignal:COMPlie:WOVerwrite {0|1|OFF|ON}
RFGSignal:COMPlie:WOVerwrite?
```

**Arguments**  
OFF or 0 disables overwriting waveform names. OFF or 0 is the default value.  
ON or 1 enables overwriting waveform names.

**Returns**  
A single <Boolean> value.

**Examples**  
```
RFGSIGNAL:COMPILE:WOVERWRITE 1 enables overwriting waveform names.
RFGSIGNAL:COMPILE:WOVERWRITE? might return 0, indicating that overwriting waveform names is disabled.
```
RFGSignal:RESet (No Query Form)

This command resets the RF Generic Signal plug-in by setting all the values within the module to their default values.

Conditions

The active plug-in must be RF Generic Signal.

Group

Control

Syntax

RFGSignal:RESet

Examples

RFGSIGNAL:RESET returns the RF Generic Signal plug-in to its default values.

RFGSignal:SPARameter:SFORmat

This command sets or returns the currently used signal format for all S-Parameter values.

Conditions

Requires an S-Parameters license.

Group

S-Parameters

Syntax

RFGSignal:SPARameter:SFORmat {RF|I|Q}
RFGSignal:SPARameter:SFORmat?

Related Commands

RFGSignal:COMPile:SFORmat

Arguments

RF sets the signal format to RF/IF and all S-Parameter settings use the RF format.
I sets the signal format to IQ and all S-Parameter settings apply to the Q signal.
Q sets the signal format to IQ and all S-Parameter settings apply to the I signal.

Returns

RF
I
Q
Examples

RFGSIGNAL:SPARAMETER:SFORMAT Q applies the S-Parameters to the Q signal.
RFGSIGNAL:SPARAMETER:SFORMAT? might return RF, indicating that all S-Parameters values are RF.
RFGSignal:SPARameter:SFORmat:LIQ

This command sets or returns the Couple Settings (I,Q) state. When enabled, all I and Q S-Parameters are linked together (chained) so that all parameters match between I and Q.

Conditions
Requires an S-Parameters license.

Group
S-Parameters

Syntax
RFGSignal:SPARameter:SFORmat:LIQ {1|0|ON|OFF}
RFGSignal:SPARameter:SFORmat:LIQ?

Related Commands
RFGSignal:SPARameter:SFORmat

Arguments
OFF or 0 disables Couple Settings. OFF or 0 is the default value.
ON or 1 enables Couple Settings.

Returns
A single <Boolean> value.

Examples
RFGSIGNAL:SPARAMETER:SFORMAT:LIQ 1 enables the Couple Settings.
RFGSIGNAL:SPARAMETER:SFORMAT:LIQ? might return 0, indicating that the Couple Settings is disabled.
RFGSignal:SPARameter:TON

This command sets or returns the S-Parameter state (enabled or disabled).

**Conditions**
Requires an S-Parameters license.

**Group**
S-Parameters

**Syntax**

RFGSignal:SPARameter:TON {1|0|ON|OFF}
RFGSignal:SPARameter:TON?

**Arguments**
OFF or 0 disables S-Parameters. OFF or 0 is the default value.
ON or 1 enables S-Parameters.

**Returns**
A single <Boolean> value.

**Examples**
RFGSIGNAL:SPARMETER:TON 1 enables the S-Parameters.
RFGSIGNAL:SPARMETER:TON? might return 0, indicating that the S-Parameters is disabled.
RFGSignal:SPARameter:BANDwidth

This command sets or returns the S-Parameter bandwidth when setting manually.

**Conditions**
Requires an S-Parameters license.

**Group**
S-Parameters

**Syntax**
RFGSignal:SPARameter:BANDwidth {FULL|<bandwidth>}
RFGSignal:SPARameter:BANDwidth?

**Related Commands**
RFGSignal:SPARameter:BANDwidth:AUTO

**Arguments**
FULL – The bandwidth is set to ½ of the waveform’s sample rate (i.e. Nyquist Frequency).

<bandwidth>::= <NR3> value.

Range: 1 Hz to ½ of the maximum sample rate of the instrument.

If the set Bandwidth is greater than the Nyquist (Sample rate of the waveform/2), then the software limits the bandwidth to ½ of the waveform’s sample rate.

**Returns**
FULL
A single <NR3> value.

**Examples**
RFGSIGNAL:SPARMETER:BANDWIDTH 60E6 sets the S-Parameter Bandwidth Value to 60 MHz.
RFGSIGNAL:SPARMETER:BANDWIDTH? might return 1.0000000000E+9, indicating the S-Parameter Bandwidth is set to 1 GHz.
**RFGSignal:SPARameter:BANDwidth:AUTO**

This command sets or returns the S-Parameter automatic bandwidth calculation setting. The bandwidth is defined at the point where the signal rolls off to –60 dB. If this results in a bandwidth greater than the instrument supports, the bandwidth is set to $\frac{1}{2}$ of the waveform’s sample rate (i.e. Nyquist Frequency).

**Conditions**
Requires an S-Parameters license.

**Group**
S-Parameters

**Syntax**

```
RFGSignal:SPARameter:BANDwidth:AUTO {0|1|OFF|ON}
RFGSignal:SPARameter:BANDwidth:AUTO?
```

**Related Commands**
RFGSignal:SPARameter:BANDwidth

**Arguments**

- **ON or 1** enables automatic bandwidth calculation. ON or 1 is the default value.
- **OFF or 0** disables automatic bandwidth calculation and sets the Bandwidth setting to Manual, requiring a value.

**Returns**
A single <Boolean> value.

**Examples**

```
RFGSIGNAL:SPARMETER:BANDWIDTH:AUTO 0 disables the S-Parameter automatic bandwidth calculation and sets it to use a manual value.
RFGSIGNAL:SPARMETER:BANDWIDTH:AUTO? might return 1, indicating the S-Parameter automatic bandwidth calculation is enabled.
```
RFGSignal:SPARameter:CASCading:AGGRessor2[:ENABle]

This command sets or returns the aggressor 2 signal type state (enabled or disabled) in Cascading mode. Aggressor2 signals are available when the number of ports is set to 12.

**Conditions**
- S-Parameter Mode must be set to Cascading.
- Number of Ports must be set to 12.
- Requires an S-Parameters license.

**Group**
- S-Parameters

**Syntax**

```
RFGSignal:SPARameter:CASCading:AGGRessor2[:ENABle] {0|1|ON|OFF}
RFGSignal:SPARameter:CASCading:AGGRessor2[:ENABle]?
```

**Related Commands**
- RFGSignal:SPARameter:MODE

**Arguments**
- OFF or 0 disables the aggressor 2 signal type. OFF or 0 is the default value.
- ON or 1 enables the aggressor 2 signal type.

**Returns**
- A single <Boolean> value.

**Examples**
- `RFGSIGNAL:SPARAMETER:CASCADING:AGGRESSOR2:ENABLE ON` enables the aggressor 2 signal type, in Cascading mode.
- `RFGSIGNAL:SPARAMETER:CASCADING:AGGRESSOR2:ENABLE?` might return 0, indicating that the aggressor 2 signal type is disabled, in Cascading mode.
**RFGSignal:SPARameter:CASCading:AGGRessor[n]:AMPLitude**

This command sets or returns the specified Aggressor's amplitude, in Cascading mode.

**Conditions**
- S-Parameter Mode must be set to Cascading.
- Number of ports must be either 8 or 12.
- Requires an S-Parameters license.

**Group**
- S-Parameters

**Syntax**
- `RFGSignal:SPARameter:CASCading:AGGRessor[n]:AMPLitude <amplitude>`
- `RFGSignal:SPARameter:CASCading:AGGRessor[n]:AMPLitude?`

**Related Commands**
- `RFGSignal:SPARameter:MODE`

**Arguments**
- `[n] ::= {1|2} (“n” determines the aggressor signal (1 or 2). 2 is only valid if the number of ports is set to 12.)
- If omitted, n is interpreted as 1.
- `<amplitude> ::= <NRf>`

**Returns**
- A single `<NR3>` value.

**Examples**
- `RFGSIGNAL:SPARAMETER:CASCADING:AGGRESSOR1:AMPLITUDE 200E-3` sets the first Aggressor's amplitude to 200 mV, in Cascading mode.
- `RFGSIGNAL:SPARAMETER:CASCADING:AGGRESSOR2:AMPLITUDE?` might return `100.0000000000E-3`, indicating that the 2nd Aggressor's amplitude is set to 100 mV, in Cascading mode.
RFGSignal:SPARameter:CASCading:AGGRessor[n]:CTALk

This command sets or returns the specified Aggressor's crosstalk type, in Cascading mode.

Conditions
S-Parameter Mode must be set to Cascading.
Number of ports must be either 8 or 12.
Requires an S-Parameters license.

Group
S-Parameters

Syntax
RFGSignal:SPARameter:CASCading:AGGRessor[n]:CTALk
{NEXT|FEXT|BOTH}
RFGSignal:SPARameter:CASCading:AGGRessor[n]:CTALK?

Related Commands
RFGSignal:SPARameter:MODE

Arguments
[n] ::= {1|2} ("n" determines the aggressor signal (1 or 2). 2 is only valid if the number of ports is set to 12.)
If omitted, n is interpreted as 1.
NEXT – Near-End Crosstalk
FEXT – Far-End Crosstalk
BOTH – Near and Far-End Crosstalk

Returns
NEXT
FEXT
BOTH

Examples
RFGSIGNAL:SPARAmeter:CASCADING:AGGRESSOR1:CTALK FEXT sets the first Aggressor's Crosstalk type to Far End Crosstalk, in Cascading mode.
RFGSIGNAL:SPARAmeter:CASCADING:AGGRESSOR2:CTALK? might return NEXT, indicating that the 2nd Aggressor crosstalk type is set to Far End Crosstalk, in Cascading mode.
RFGSignal:SPARameter:CASCading:AGGReSSor[n]:DRATe

This command sets or returns the specified Aggressor's data rate, in Cascading mode.

**Conditions**

- S-Parameter Mode must be set to Cascading.
- Number of ports must be either 8 or 12.
- Requires an S-Parameters license.

**Group**

S-Parameters

**Syntax**

RFGSignal:SPARameter:CASCading:AGGReSSor[n]:DRATe

<data_rate>

RFGSignal:SPARameter:CASCading:AGGReSSor[n]:DRATe?

**Related Commands**

RFGSignal:SPARameter:MODE

**Arguments**

[n] ::= {1|2} (“n” determines the aggressor signal (1 or 2). 2 is only valid if the number of ports is set to 12.)

If omitted, n is interpreted as 1.

<data_rate> ::= <NRf>

**Returns**

A single <NR3> value.

**Examples**

RFGSIGNAL:SPARAMETER:CASCADING:AGGRESSOR1:DRATE 4E9 sets the first Aggressor's data rate to 4 Gbps, in Cascading mode.

RFGSIGNAL:SPARAMETER:CASCADING:AGGRESSOR2:DRATE? might return 500.0000000000E+3, indicating that the 2nd Aggressor's data rate is set to 500 kbps, in Cascading mode.
RFGSignal:SPARameter:CASCading:AGGressor[n]:SIGNal

This command sets or returns specified Aggressor's signal type, in Cascading mode.

**Conditions**
S-Parameter Mode must be set to Cascading.
Number of ports must be either 8 or 12.
Requires an S-Parameters license.

**Group**
S-Parameters

**Syntax**
RFGSignal:SPARameter:CASCading:AGGRessor[n]:SIGNal {CLOCk|PRBS|FILE|SAVictim}
RFGSignal:SPARameter:CASCading:AGGRessor[n]:SIGNal?

**Related Commands**
RFGSignal:SPARameter:MODE
RFGSignal:SPARameter:CASCading:AGGRessor[n]:SIGNal:FILE

**Arguments**
[n] ::= {1|2} (“n” determines the aggressor signal (1 or 2). 2 is only valid if the number of ports is set to 12.)

If omitted, n is interpreted as 1.

CLOCK – Indicates that the aggressor signal is a clock pattern.
PRBS – Indicates that the aggressor signal is a PBRS pattern. You also must set the PBRS type.
FILE – Aggressor is set to use a file. You must set the file path.
SAVictim – Aggressor is the same as the victim.

**Returns**
CLOC
PRBS
FILE
SAV

**Examples**
RFGSIGNAL:SPARAMETER:CASCADING:AGGRESSOR1:SIGNAL SAVICTIM sets the aggressor signal to be the same as the victim, in Cascading mode.

RFGSIGNAL:SPARAMETER:CASCADING:AGGRESSOR2:SIGNAL? might return FILE, indicating that 2nd Aggressor has a signal type set to use a file, in Cascading mode.
**RFGSignal:SPARameter:CASCading:AGGressor[n]:SIGNal:FILE**

This command sets or returns the filepath to the aggressor file for the specified Aggressor, in Cascading mode.

**Conditions**
- S-Parameter Mode must be set to Cascading.
- Number of ports must be either 8 or 12.
- Aggressor signal type must be File.
- Requires an S-Parameters license.

**Group**
- S-Parameters

**Syntax**

RFGSignal:SPARameter:CASCading:AGGressor[n]:SIGNal:FILE

<filepath>

RFGSignal:SPARameter:CASCading:AGGressor[n]:SIGNal:FILE?

**Related Commands**

- RFGSignal:SPARameter:MODE
- RFGSignal:SPARameter:CASCading:AGGressor[n]:SIGNal

**Arguments**

[n] ::= {1|2} ("n" determines the aggressor signal (1 or 2). 2 is only valid if the number of ports is set to 12.)

If omitted, n is interpreted as 1.

<filepath> ::= <string> defines the path to the aggressor file.

**Returns**

A single <filepath> string.

**Examples**

RFGSIGNAL:SPARAMETER:CASCADING:AGGRESSOR1:SIGNAL:FILE
"C:\temp\myFile.txt" sets the first Aggressor's file and filepath when the aggressor is set to use a file, in Cascading mode.

RFGSIGNAL:SPARAMETER:CASCADING:AGGRESSOR2:SIGNAL:FILE?

might return "C:\temp\myFile.txt" indicating that the 2nd Aggressor has a signal type filepath set to "C:\temp\myFile.txt", in Cascading mode.
**RFGSignal:SPARameter:CASCading:AGGRessor[n]:SIGNal:PRBS**

This command sets or returns the specified Aggressor's PRBS signal type, in Cascading mode.

**Conditions**
- S-Parameter Mode must be set to Cascading.
- Number of ports must be either 8 or 12.
- Aggressor signal type must be PRBS.
- Requires an S-Parameters license.

**Group**
- **S-Parameters**

**Syntax**

```
RFGSignal:SPARameter:CASCading:AGGRessor[n]:SIGNal:PRBS
{PRBS7|PRBS9|PRBS15|PRBS16|PRBS20|PRBS21|PRBS23|PRBS29|PRBS31}
```

```
RFGSignal:SPARameter:CASCading:AGGRessor[n]:SIGNal:PRBS?
```

**Related Commands**
- RFGSignal:SPARameter:MODE
- RFGSignal:SPARameter:CASCading:AGGRessor[n]:SIGNal

**Arguments**
- `[n] ::= {1|2}` (“n” determines the aggressor signal (1 or 2). 2 is only valid if the number of ports is set to 12.)
  - If omitted, `n` is interpreted as 1.
- Patterns available include: PRBS7, PRBS9, PRBS15, PRBS16, PRBS20, PRBS21, PRBS23, PRBS29, PRBS31.

**Returns**
- PRBS7, PRBS9, PRBS15, PRBS16, PRBS20, PRBS21, PRBS23, PRBS29, PRBS31

**Examples**

```
RFGSIGNAL:SPARAMETER:CASCADING:AGGRESSOR1:SIGNAL:PRBS PRBS31
```
sets the first Aggressor's Signal type's PRBS value to PRBS31, in Cascading mode.

```
RFGSIGNAL:SPARAMETER:CASCADING:AGGRESSOR2:SIGNAL:PRBS?
```
might return `PRBS15`, indicating that the 2nd Aggressor has a signal type PRBS value set to PRBS15, in Cascading mode.
RFGSignal:SPARameter:CASCading:DEEMbed

This command sets or returns whether the Cascading S-Parameters is to de-embed (invert) the S-Parameters, in Cascading mode.

**Conditions**  
S-Parameter Mode must be set to Cascading.  
Requires an S-Parameters license.

**Group**  
S-Parameters

**Syntax**  
RFGSignal:SPARameter:CASCading:DEEMbed \{0|1|OFF|ON\}  
RFGSignal:SPARameter:CASCading:DEEMbed?

**Related Commands**  
RFGSignal:SPARameter:MODE

**Arguments**  
OFF or 0 disables de-embedding. OFF or 0 is the default value.  
ON or 1 enables de-embedding.

**Returns**  
A single <Boolean> value.

**Examples**  
RFGSIGNAL:SPARAMETER:CASCADING:DEEMBED 1 will de-embed the S-Parameters for Cascading mode.  
RFGSIGNAL:SPARAMETER:CASCADING:DEEMBED? might return 0, indicating that S-Parameters will not be de-embedded for Cascading mode.
RFGSignal:SPARameter:CASCading:STAGE[m]:DRX[n]

This command sets or returns the S-Parameter port assignment of the specified Stage and the channel’s specified receiver port number (Rx-Port) in Cascading mode and Differential Signalling Scheme (where applicable).

**Conditions**
- S-Parameter Mode must be set to Cascading.
- S-Parameter Signalling Scheme must be set to Differential (where applicable).
- Requires an S-Parameters license.

**Group**
S-Parameters

**Syntax**
RFGSignal:SPARameter:CASCading:STAGE[m]:DRX[n] <port number>
RFGSignal:SPARameter:CASCading:STAGE[m]:DRX[n]?

**Related Commands**
- RFGSignal:SPARameter:MODE
- RFGSignal:SPARameter:CASCading:STYPe
- RFGSignal:SPARameter:CASCading:TYPE
- RFGSignal:SPARameter:CASCading:STAGE[m]:DTX[n]

**Arguments**

[m] ::= {1|2|3|4|5|6}. A variable value to define the Stage.

If omitted, interpreted as 1

[n] ::= <NR1> value. A variable value to define the receiver port number (Rx-Port) of the channel.

The actual range is dependent on the Number of Ports (Type).

Type = 4, then n = {1}
Type = 8 then n = {1 – 2}
Type = 12 then n = {1 – 3}

If omitted, n is interpreted as 1.

<port number> ::= <NR1>. A variable value to define the S-Parameter Port assigned to the specified Rx-Port of the channel.

The actual range is dependent on the Number of Ports (Type).

Type = 4 then <port number> = {1 – 2}
Type = 8 then <port number> = {1 – 4}
Type = 12 then <port number> = {1 – 6}

**Returns**
A single <NR1> value.
Examples

`RFGSIGNAL:SPARAMETER:CASCADING:STAGE2:DRX2 <params>` assigns S-Parameter port 4 to the channel’s receiver port 2 for Stage 2, in the Differential, Cascading mode.

`RFGSIGNAL:SPARAMETER:CASCADING:STAGE6:RX3?` might return 10, indicating that S-Parameter Port 10 is assigned to the channel’s receiver port 3 for Stage 6, in the Differential, Cascading mode.
RFGSignal:SPARameter:CASCading:STAGe[m]:DTX[n]

This command sets or returns the S-Parameter port assignment of the specified Stage and the channel’s specified transmission port number (Tx-Port) in Cascading mode and Differential Signalling Scheme (where applicable).

Conditions
S-Parameter Mode must be set to Cascading.
S-Parameter Signalling Scheme must be set to Differential (where applicable).
Requires an S-Parameters license.

Group
S-Parameters

Syntax
RFGSignal:SPARameter:CASCading:STAGe[m]:DTX[n] <port number>
RFGSignal:SPARameter:CASCading:STAGe[m]:DTX[n]?  

Related Commands
RFGSignal:SPARameter:MODE
RFGSignal:SPARameter:CASCading:STYPe
RFGSignal:SPARameter:CASCading:TYPE
RFGSignal:SPARameter:CASCading:STAGe[m]:DRX[n]

Arguments
[m] ::= {1|2|3|4|5|6}. A variable value to define the Stage.
If omitted, interpreted as 1
[n] ::= <NR1> value. A variable value to define the transmission port number (Tx-Port) of the channel.
The actual range is dependent on the Number of Ports (Type).
Type = 4, then n = {1}  
Type = 8 then n = {1 – 2} 
Type = 12 then n = {1 – 3} 
If omitted, n is interpreted as 1.

<port number> ::= <NR1>. A variable value to define the S-Parameter Port assigned to the specified Tx-Port of the channel.
The actual range is dependent on the Number of Ports (Type).
Type = 4 then <port number> = {1 – 2} 
Type = 8 then <port number> = {1 – 4}  
Type = 12 then <port number> = {1 – 6}

Returns
A single <NR1> value.
**Examples**

`RFGSIGNAL:SPARAMETER:CASCADING:STAGE2:DTX2` 4 assigns S-Parameter port 4 to the channel’s transmission port 2 for Stage 2, in the Differential, Cascading mode.

`RFGSIGNAL:SPARAMETER:CASCADING:STAGE6:TX3?` might return 10, indicating that S-Parameter Port 10 is assigned to the channel’s transmission port 3 for Stage 6, in the Differential, Cascading mode.
**RFGSignal:SPARameter:CASCading:STAGE[m]:ENABle**

This command sets or returns the state of the specified Cascaded S-Parameter stage (enabled or disabled).

**Conditions**
S-Parameter Mode must be set to Cascading.
Requires an S-Parameters license.

**Group**
S-Parameters

**Syntax**
`RFGSignal:SPARameter:CASCading:STAGE[m]:ENABle {0|1|OFF|ON}`
`RFGSignal:SPARameter:CASCading:STAGE[m]:ENABle?`

**Related Commands**
`RFGSignal:SPARameter:MODE`
`RFGSignal:SPARameter:NCASCading:FILE`

**Arguments**

\[m\] ::= \{1 – 6\}  ("m" determines the stage number)

If omitted, m is interpreted as 1.
OFF or 0 disables the specified Cascading Stage. OFF or 0 is the default value.
ON or 1 enables the Stage.

**Returns**
A single <Boolean> value.

**Examples**

`RFGSIGNAL:SPARAMETER:CASCADING:STAGE6:ENABLE 1` enables Stage 6 in Cascading mode.

`RFGSIGNAL:SPARAMETER:CASCADING:STAGE6:ENABLE?` might return 0, indicating that Stage 6 is not enabled in Cascading mode.
RFGSignal:SPARameter:CASCading:STAGE[m]:FILE

This command sets or returns the filepath for the specified S-Parameters Cascading Stage, in Cascading mode.

Conditions
S-Parameter Mode must be set to Cascading.
Requires an S-Parameters license.

Group
S-Parameters

Syntax
RFGSignal:SPARameter:CASCading:STAGE[m]:FILE <filepath>

Related Commands
RFGSignal:SPARameter:MODE

Arguments
[m] ::= \{1 – 6\} ("m" determines the stage number)
If omitted, m is interpreted as 1.

Returns
<filepath>::= <string>.

Examples
RFGSIGNAL:SPARAMETER:CASCADING:STAGE1:FILE
"C:\temp\myFile.s12p" sets the filepath to "C:\temp\myFile.s12p" for use during compilation for Stage 1.

RFGSignal:SPARameter:CASCading:STAGe[m]:RX[n]

This command sets or returns the S-Parameter port assignment of the specified Stage and the channel's specified receiver port number (Rx-Port) in Cascading mode and Single-Ended Signalling Scheme (where applicable).

Conditions
S-Parameter Mode must be set to Cascading.
S-Parameter Signalling Scheme must be set to Single-Ended (where applicable).
Requires an S-Parameters license.

Group
S-Parameters

Syntax
RFGSignal:SPARameter:CASCading:STAGe[m]:RX[n] <port_number>
RFGSignal:SPARameter:CASCading:STAGe[m]:RX[n]?

Related Commands
RFGSignal:SPARameter:MODE
RFGSignal:SPARameter:CASCading:STYPe
RFGSignal:SPARameter:CASCading:TYPE
RFGSignal:SPARameter:CASCading:STAGe[m]:TX[n]

Arguments

[m] ::= {1|2|3|4|5|6}. A variable value to define the Stage.
If omitted, interpreted as 1

[n] ::= <NR1> value. A variable value to define the receiver port number (Rx-Port) of the channel.
The actual range is dependent on the Number of Ports (Type).
Type = 1, then n = no value
Type = 2, then n = {1}
Type = 4 then n = {1 – 2}
Type = 6 then n = {1 – 3}
Type = 8 then n = {1 – 4}
Type = 12 then n = {1 – 6}

<port_number> ::= <NR1>. A variable value to define the S-Parameter Port assigned to the specified Tx-Port of the channel.
The actual range is dependent on the Number of Ports (Type).
Type = 1, then <port_number> = no value
Type = 2, then <port_number> = {1 – 2}
Type = 4 then <port_number> = {1 – 4}
Type = 6 then <port_number> = {1 – 6}
Type = 8 then <port number> = {1 – 8}
Type = 12 then <port number> = {1 – 12}

**Returns**
A single <NR1> value.

**Examples**

RFGSIGNAL:SPARAMETER:CASCADING:STAGE2:RX2 4 assigns S-Parameter port 4 to the channel’s receiver port 2 for Stage 2, in the Single-Ended, Cascading mode.

RFGSIGNAL:SPARAMETER:CASCADING:STAGE6:RX3? might return 10, indicating that S-Parameter Port 10 is assigned to the channel’s receiver port 3 for Stage 6, in the Single-Ended, Cascading mode.
RFGSignal:SPARameter:CASCading:STAGe[m]:SSCHeme

This command sets or returns the S-Parameter Signalling Scheme, in Cascading mode. Signalling Scheme is only available when the Number of Ports is set to 4, 8, or 12.

Conditions

- S-Parameter Mode must be set to Cascading.
- Requires an S-Parameters license.

Group

S-Parameters

Syntax

\[
\text{RFGSignal:SPARameter:CASCading:STAGe[m]:SSCHeme} \{\text{SENDed|DIFFerential}\} \\
\text{RFGSignal:SPARameter:CASCading:STAGe[m]:SSCHeme?}
\]

Related Commands

RFGSignal:SPARameter:MODE

Arguments

\[
[m] ::= \{1 – 6\} \text{ ("m" determines the stage number)}
\]

- If omitted, m is interpreted as 1.
- SENDed – Single Ended Signal Scheme
- DIFFerential – Differential Signal Scheme

Returns

- SEND
- DIFF

Examples

\[
\text{RFGSIGNAL:SPARAMETER:CASCADING:STAGE2:SSCHEME DIFF} \text{ sets the Stage 2 Signalling Scheme to Differential, in Cascading mode.}
\]

\[
\text{RFGSIGNAL:SPARAMETER:CASCADING:STAGE3:SSCHEME? might return SEND, indicating that the Stage 3 Signalling Scheme is set to Single Ended, in Cascading mode.}
\]
**RFGSignal:SPARameter:CASCading:STAGe[m]:TX[n]**

This command sets or returns the S-Parameter port assignment of the specified Stage and the channel’s specified transmission port number (Tx-Port) in Cascading mode and Single-Ended Signalling Scheme (where applicable).

**Conditions**
- S-Parameter Mode must be set to Cascading.
- S-Parameter Signalling Scheme must be set to Single-Ended (where applicable).
- Requires an S-Parameters license.

**Group**
S-Parameters

**Syntax**
```
RFGSignal:SPARameter:CASCading:STAGe[m]:TX[n] <port number>
RFGSignal:SPARameter:CASCading:STAGe[m]:TX[n]?
```

**Related Commands**
- RFGSignal:SPARameter:MODE
- RFGSignal:SPARameter:CASCading:STYPe
- RFGSignal:SPARameter:CASCading:TYPE
- RFGSignal:SPARameter:CASCading:STAGe[m]:RX[n]

**Arguments**
- `[m] ::= \{1|2|3|4|5|6\}`. A variable value to define the Stage.
  - If omitted, interpreted as 1
- `[n] ::= <NR1> value`. A variable value to define the transmission port number (Tx-Port) of the channel.
  - The actual range is dependent on the Number of Ports (Type).
    - Type = 1, then `n = no value`
    - Type = 2, then `n = \{1\}`
    - Type = 4 then `n = \{1 – 2\}`
    - Type = 6 then `n = \{1 – 3\}`
    - Type = 8 then `n = \{1 – 4\}`
    - Type = 12 then `n = \{1 – 6\}`
- `<port number> ::= <NR1>`. A variable value to define the S-Parameter Port assigned to the specified Tx-Port of the channel.
  - The actual range is dependent on the Number of Ports (Type).
    - Type = 1, then `<port number> = no value`
    - Type = 2, then `<port number> = \{1 – 2\}`
    - Type = 4 then `<port number> = \{1 – 4\}`
    - Type = 6 then `<port number> = \{1 – 6\}`
Type = 8 then <port number> = \{1 – 8\}
Type = 12 then <port number> = \{1 – 12\}

**Returns**

A single <NR1> value.

**Examples**

`RFGSIGNAL:SPARAMETER:CASCADING:STAGE2:TX2 4` assigns S-Parameter port 4 to the channel’s transmission port 2 for Stage 2, in the Single-Ended, Cascading mode.

`RFGSIGNAL:SPARAMETER:CASCADING:STAGE6:TX3?` might return 10, indicating that S-Parameter Port 10 is assigned to the channel’s transmission port 3 for Stage 6, in the Single-Ended, Cascading mode.
**RFGSignal:SPARameter:CASCading:STYPe**

This command sets or returns S-Parameter signal type (victim or aggressor), in Cascading mode. The number of ports must be either 8 or 12.

**Conditions**
- S-Parameter Mode must be set to Cascading.
- Number of Ports must be either 8 or 12.
- Requires an S-Parameters license.

**Group**
- S-Parameters

**Syntax**

```
RFGSignal:SPARameter:CASCading:STYPe {VICTim|AGGRessor|BOTH}
RFGSignal:SPARameter:CASCading:STYPe?
```

**Related Commands**
- **RFGSignal:SPARameter:MODE**

**Arguments**
- VICTim – enables the victim signal type.
- AGGRessor – enables the aggressor signal type.
- BOTH – enables the victim and aggressor signal types.

**Returns**
- VICT
- AGGR
- BOTH

**Examples**

```
RFG SIGNAL:SPARAMETER:CASCADING:STYPE BOTH sets the signal type to include both the Victim and Aggressor signal types, in Cascading mode.

RFG SIGNAL:SPARAMETER:CASCADING:STYPE? might return AGGR, indicating that the S-Parameter signal type is currently set to be Aggressor, in Cascading mode.
```
**RFGSignal:SPARameter:CASCading:TYPE**

This command sets or returns the S-Parameter number of ports, in Cascading mode.

**Conditions**
Requires an S-Parameters license.

**Group**
S-Parameters

**Syntax**

```
RFGSignal:SPARameter:CASCading:TYPE {2|4|6|8|12}
RFGSignal:SPARameter:CASCading:TYPE?
```

**Arguments**

{2|4|6|8|12} – defines the number of S-Parameter ports.

**Returns**
A single <NR1> value.

**Examples**

```
RFGSIGNAL:SPARAMETER:CASCADING:TYPE 12 sets the S-Parameter type to a 12-Port system for the cascading mode.

RFGSIGNAL:SPARAMETER:CASCADING:TYPE? might return 6, indicating that the S-Parameter type is a 6-Port system for Cascading mode.
```
RFGSignal:SPARameter:MODE

This command sets or returns the S-Parameter mode (Cascading or Non-Cascading).

**Conditions**
Requires an S-Parameters license.

**Group**
S-Parameters

**Syntax**
RFGSignal:SPARameter:MODE {CASC|NCAS}
RFGSignal:SPARameter:MODE?

**Arguments**
CASCading sets the S-Parameter mode to cascading, allowing you to cascade up to six S-parameter files and apply the characteristics on the waveform.

NCASCading sets the S-Parameter mode to non-cascading, allowing you to apply S-parameter characteristics on the waveform from only one S-parameter file.

**Returns**
CASC
NCASC

**Examples**
RFGSIGNAL:SPARAMETER:MODE CASCADING sets the S-Parameter mode to cascading.

RFGSIGNAL:SPARAMETER:MODE? might return NCAS, indicating that the S-Parameter mode is set to Non-Cascading mode.
RFGSignal:SPARameter:NCAScading:AGGRessor2[:ENABLE]

This command sets or returns the aggressor 2 signal type state (enabled or disabled) in Non-Cascading mode. Aggressor2 signals are available when the number of ports is set to 12.

Conditions
- S-Parameter Mode must be set to Non-Cascading.
- Number of Ports must be set to 12.
- Requires an S-Parameters license.

Group
- S-Parameters

Syntax
- \texttt{RFGSignal:SPARameter:NCAScading:AGGRessor2[:ENABLE]}
  - \{0|1|ON|OFF\}
- \texttt{RFGSignal:SPARameter:NCAScading:AGGRessor2[:ENABLE]?}

Related Commands
- \texttt{RFGSignal:SPARameter:MODE}

Arguments
- OFF or 0 disables the aggressor 2 signal type. OFF or 0 is the default value.
- ON or 1 enables the aggressor 2 signal type.

Returns
- A single <Boolean> value.

Examples
- \texttt{RFGSIGNAL:SPARAMETER:NCASCADING:AGGRESSOR2:ENABLE} ON enables the aggressor 2 signal type, in Non-Cascading mode.
- \texttt{RFGSIGNAL:SPARAMETER:NCASCADING:AGGRESSOR2:ENABLE?} might return 0, indicating that the aggressor 2 signal type is disabled, in Non-Cascading mode.
**RFGSignal:SPARameter:NCAScading:AGGRessor[n]:AMPLitude**

This command sets or returns the specified Aggressor's amplitude, in Non-Cascading mode.

**Conditions**
- S-Parameter Mode must be set to Non-Cascading.
- Number of ports must be either 8 or 12.
- Requires an S-Parameters license.

**Group**
- S-Parameters

**Syntax**

RFGSignal:SPARameter:NCAScading:AGGRessor[n]:AMPLitude

RFGSignal:SPARameter:NCAScading:AGGRessor[n]:AMPLitude?

**Related Commands**
- **RFGSignal:SPARameter:MODE**

**Arguments**

[n] ::= {1|2}  (“n” determines the aggressor signal (1 or 2). 2 is only valid if the number of ports is set to 12.)

If omitted, n is interpreted as 1.

<amplitude>::= <NR3> value.

**Returns**
- A single <NR3> value.

**Examples**

RFGSIGNAL:SPARAMETER:NCASCADING:AGGRESSOR1:AMPLITUDE 200E-3
sets the 1st Aggressor's amplitude to 200 mV, in Non-Cascading mode.

RFGSIGNAL:SPARAMETER:NCASCADING:AGGRESSOR2:AMPLITUDE?
might return 100.0000000000E-3, indicating that the 2nd Aggressor's amplitude is set to 100 mV, in Non-Cascading mode.
RFGSignal:SPARameter:NCAScading:AGGRessor[n]:CTALk

This command sets or returns the specified Aggressor's crosstalk type, in Non-Cascading mode.

**Conditions**

- S-Parameter Mode must be set to Non-Cascading.
- Number of ports must be either 8 or 12.
- Requires an S-Parameters license.

**Group**

S-Parameters

**Syntax**

```
RFGSignal:SPARameter:NCAScading:AGGRessor[n]:CTALk
{NEXT|FEXT|BOTH}
RFGSignal:SPARameter:NCAScading:AGGRessor[n]:CTALk?
```

**Related Commands**

RFGSignal:SPARameter:MODE

**Arguments**

\[n\] ::= \{1|2\} (“n” determines the aggressor signal (1 or 2). 2 is only valid if the number of ports is set to 12.)

If omitted, n is interpreted as 1.

- NEXT – Near-End Crosstalk
- FEXT – Far-End Crosstalk
- BOTH – Near and Far-End Crosstalk

**Returns**

- NEXT
- FEXT
- BOTH

**Examples**

```
RFGSIGNAL:SPARAMETER:NCASCADING:AGGRESSOR1:CTALK FEXT sets the 1st Aggressor's Crosstalk type to Far End Crosstalk, in Non-Cascading mode.

RFGSIGNAL:SPARAMETER:NCASCADING:AGGRESSOR2:CTALK? might return NEXT, indicating that the 2nd Aggressor crosstalk type is set to Near End Crosstalk, in Non-Cascading mode.
```
**RFGSignal:SPARameter:NCAScading:AGGRessor[n]:DRATe**

This command sets or returns the specified Aggressor's data rate, in Non-Cascading mode.

**Conditions**
- S-Parameter Mode must be set to Non-Cascading.
- Number of ports must be either 8 or 12.
- Requires an S-Parameters license.

**Group**
- S-Parameters

**Syntax**

- `RFGSignal:SPARameter:NCAScading:AGGRessor[n]:DRATe`<data_rate>
- `RFGSignal:SPARameter:NCAScading:AGGRessor[n]:DRATe?`

**Related Commands**
- `RFGSignal:SPARameter:MODE`

**Arguments**

- `[n] ::= {1|2} (“n” determines the aggressor signal (1 or 2). 2 is only valid if the number of ports is set to 12.)
- If omitted, n is interpreted as 1.
- `<data_rate>::= <NR3> value.`

**Returns**
- A single `<NR3>` value.

**Examples**

- `RFGSIGNAL:SPARAMETER:NCASCADING:AGGRESSOR1:DRATE 4E9` sets the 1st Aggressor's data rate to 4 Gbps, in Non-Cascading mode.
- `RFGSIGNAL:SPARAMETER:NCASCADING:AGGRESSOR2:DRATE?` might return `500.0000000000E+3`, indicating that the 2nd Aggressor's data rate is set to 500 kbps, in Non-Cascading mode.
**RFGSignal:SPARameter:NCAScading:AGGRessor[n]:SIGNal**

This command sets or returns specified Aggressor's signal type, in Non-Cascading mode.

**Conditions**

S-Parameter Mode must be set to Non-Cascading.

Number of ports must be either 8 or 12.

Requires an S-Parameters license.

**Group**

S-Parameters

**Syntax**

RFGSignal:SPARameter:NCAScading:AGGRessor[n]:SIGNal

{CLOCK|PRBS|FILE|SAVictim}

RFGSignal:SPARameter:NCAScading:AGGRessor[n]:SIGNal?

**Related Commands**

RFGSignal:SPARameter:MODE

RFGSignal:SPARameter:CASCading:AGGRessor[n]:SIGNal:FILE

**Arguments**

[n] ::= {1|2} (“n” determines the aggressor signal (1 or 2). 2 is only valid if the number of ports is set to 12.)

If omitted, n is interpreted as 1.

CLOCK – Indicates that the aggressor signal is a clock pattern.

PRBS – Indicates that the aggressor signal is a PBRS pattern. You also must set the PBRS type.

FILE – Aggressor is set to use a file. You must set the file path.

SAVictim – Aggressor is the same as the victim.

**Returns**

CLOC

PRBS

FILE

SAV

**Examples**

RFGSIGNAL:SPARAMETER:NCASCADING:AGGRESSOR1:SIGNAL SAVICTIM

sets the 1st aggressor signal to be the same as the victim, in Non-Cascading mode

RFGSIGNAL:SPARAMETER:NCASCADING:AGGRESSOR2:SIGNAL? might return FILE, indicating that 2nd Aggressor has a signal type set to use a file, in Non-Cascading mode.
RFGSignal:SPARameter:NCAScading:AGGRessor[n]:SIGNal:FILE

This command sets or returns the filepath to the aggressor file for the specified Aggressor, in Non-Cascading mode.

**Conditions**
- S-Parameter Mode must be set to Non-Cascading.
- Number of ports must be either 8 or 12.
- Aggressor signal type must be File.
- Requires an S-Parameters license.

**Group**
S-Parameters

**Syntax**

```
RFGSignal:SPARameter:NCAScading:AGGRessor[n]:SIGNal:FILE
<filepath>
RFGSignal:SPARameter:NCAScading:AGGRessor[n]:SIGNal:FILE?
```

**Related Commands**
- RFGSignal:SPARameter:MODE
- RFGSignal:SPARameter:NCAScading:AGGRessor[n]:SIGNal

**Arguments**

`[n] ::= \{1|2\}` ("n" determines the aggressor signal (1 or 2). 2 is only valid if the number of ports is set to 12.)

If omitted, n is interpreted as 1.

`<filepath>::= <string>` defines the path to the aggressor file.

**Returns**
A single `<filepath>` string.

**Examples**

```
RFGSIGNAL:SPARMETER:NCASCADING:AGGRESSOR1:SIGNAL:FILE
"C:\temp\myFile.txt" sets the 1st Aggressor's file and filepath when the aggressor is set to use a file, in Non-Cascading mode.

RFGSIGNAL:SPARMETER:NCASCADING:AGGRESSOR2:SIGNAL:FILE? might return "C:\temp\myFile.txt" indicating that the 2nd Aggressor has a signal type filepath set to "C:\temp\myFile.txt", in Non-Cascading mode.
```
**RFGSignal:SPARameter:NCAScading:AGGRessor[n]:SIGNal:PRBS**

This command sets or returns the specified Aggressor's PRBS signal type, in Non-Cascading mode.

**Conditions**

- S-Parameter Mode must be set to Non-Cascading.
- Number of ports must be either 8 or 12.
- Aggressor signal type must be PRBS.
- Requires an S-Parameters license.

**Group**

S-Parameters

**Syntax**

```
RFGSignal:SPARameter:NCAScading:AGGRessor[n]:SIGNal:PRBS
{PRBS7|PRBS9|PRBS15|PRBS16|PRBS20|PRBS21|PRBS23|PRBS29|PRBS31}
RFGSignal:SPARameter:NCAScading:AGGRessor[n]:SIGNal:PRBS?
```

**Related Commands**

- RFGSignal:SPARameter:MODE
- RFGSignal:SPARameter:NCAScading:AGGRessor[n]:SIGNal

**Arguments**

\[n\] ::= {1|2} (“n” determines the aggressor signal (1 or 2). 2 is only valid if the number of ports is set to 12.)

If omitted, \(n\) is interpreted as 1.

Patterns available include: PRBS7, PRBS9, PRBS15, PRBS16, PRBS20, PRBS21, PRBS23, PRBS29, PRBS31

**Returns**

PRBS7, PRBS9, PRBS15, PRBS16, PRBS20, PRBS21, PRBS23, PRBS29, PRBS31

**Examples**

```
RFGSIGNAL:SPARAMETER:NCASCADING:AGRESSOR1:SIGNAL:PRBS
PRBS31
```

sets the 1st Aggressor's Signal type's PRBS value to PRBS31, in Non-Cascading mode.

```
RFGSIGNAL:SPARAMETER:NCASCADING:AGRESSOR2:SIGNAL:PRBS?
```

might return PRBS15, indicating that the 2nd Aggressor has a signal type PRBS value set to PRBS15, in Non-Cascading mode.
**RFGSignal:SPARameter:NCAScading:DEEMbed**

This command sets or returns whether the Non-Cascading S-Parameters is to de-embed (invert) the S-Parameters, in Non-Cascading mode.

**Conditions**
- S-Parameter Mode must be set to Non-Cascading.
- Requires an S-Parameters license.

**Group**
- S-Parameters

**Syntax**
- `RFGSignal:SPARameter:NCAScading:DEEMbed {0|1|OFF|ON}`
- `RFGSignal:SPARameter:NCAScading:DEEMbed?`

**Related Commands**
- `RFGSignal:SPARameter:MODE`

**Arguments**
- OFF or 0 disables de-embedding. OFF or 0 is the default value.
- ON or 1 enables de-embedding.

**Returns**
- A single <Boolean> value.

**Examples**
- `RFGSIGNAL:SPARAMETER:NCASCADING:DEEMBED 1` will de-embed the S-Parameters for Non-Cascading mode.
- `RFGSIGNAL:SPARAMETER:NCASCADING:DEEMBED?` might return 0, indicating that S-Parameters will not be de-embedded for Non-Cascading mode.
RFGSignal:SPARameter:NCAScading:DRX[n]

This command sets or returns the S-Parameter port assignment of the channel’s specified receiver port number (Rx-Port) in Non-Cascading mode and Differential Signalling Scheme (where applicable).

**Conditions**
- S-Parameter Mode must be set to Non-Cascading.
- S-Parameter Signalling Scheme must be set to Differential.
- Requires an S-Parameters license.

**Group**
- S-Parameters

**Syntax**

RFGSignal:SPARameter:NCAScading:DRX[n] <port_number>

RFGSignal:SPARameter:NCAScading:DRX[n]?

**Related Commands**
- RFGSignal:SPARameter:MODE
- RFGSignal:SPARameter:CASCading:STYPe
- RFGSignal:SPARameter:NCAScading:TYPE
- RFGSignal:SPARameter:NCAScading:RX[n]

**Arguments**

[n] ::= <NR1>. A variable value to define the receiver port number (Rx-Port) of the channel.

The actual range is dependent on the Number of Ports (Type).

- Type = 4, then n = {1}
- Type = 8 then n = {1 – 2}
- Type = 12 then n = {1 – 3}

If omitted, n is interpreted as 1.

<port_number>::= <NR1> value. A variable value to define the S-Parameter Port assigned to the specified Rx-Port of the channel.

The actual range is dependent on the Number of Ports (Type).

- Type = 4, then <port_number> = {1 – 2}
- Type = 8 then <port_number> = {1 – 4}
- Type = 12 then <port_number> = {1 – 6}

**Returns**
- A single <NR1> value.

**Examples**
- RFGSIGNAL:SPARAMETER:NCASCADING:DTX2 4 assigns S-Parameter port 4 to channel’s receiver port 2, in the Differential, Non-Cascading mode.
RFGSIGNAL:SPARAMETER:NCASCADING:DTX3? might return 6, indicating that S-Parameter Port 6 is assigned to the channel’s receiver port 3, in the Differential, Non-Cascading mode.
RFGSignal:SPARameter:NCAScading:DTX[n]

This command sets or returns the S-Parameter port assignment of the channel’s specified transmission port number (Tx-Port) in Non-Cascading mode and Differential Signalling Scheme (where applicable).

**Conditions**

- S-Parameter Mode must be set to Non-Cascading.
- S-Parameter Signalling Scheme must be set to Differential.
- Requires an S-Parameters license.

**Group**

S-Parameters

**Syntax**

RFGSignal:SPARameter:NCAScading:DTX[n] <port_number>

RFGSignal:SPARameter:NCAScading:DTX[n]?

**Related Commands**

- RFGSignal:SPARameter:MODE
- RFGSignal:SPARameter:NCAScading:STYPE
- RFGSignal:SPARameter:NCAScading:TYPE
- RFGSignal:SPARameter:NCAScading:DRX[n]

**Arguments**

[n] ::= <NR1> value. A variable value to define the transmission port number (Tx-Port) of the channel.

The actual range is dependent on the Number of Ports (Type).

- Type = 4, then n = {1}
- Type = 8 then n = {1 – 2}
- Type = 12 then n = {1 – 3}

If omitted, n is interpreted as 1.

<port_number> ::= <NR1> value. A variable value to define the S-Parameter Port assigned to the specified Tx-Port of the channel.

The actual range is dependent on the Number of Ports (Type).

- Type = 4 then <port number> = {1 – 2}
- Type = 8 then <port number> = {1 – 4}
- Type = 12 then <port number> = {1 – 6}

**Returns**

A single <NR1> value.

**Examples**

RFGSIGNAL:SPARAMETER:NCASCADING:DTX2 4 assigns S-Parameter port 4 to channel’s transmission port 2, in the Differential, Non-Cascading mode.
RF SGNL:SPRMTR:NCASC:DTOX3? might return 6, indicating that S-Parameter Port 6 is assigned to the channel’s transmission port 3, in the Differential, Non-Cascading mode.
RFGSignal:SPARameter:NCAScading:FILE

This command sets or returns the filepath and file name of the S-Parameter file, in Non-Cascading mode.

Conditions
S-Parameter Mode must be set to Non-Cascading.
Requires an S-Parameters license.

Group
S-Parameters

Syntax
RFGSignal:SPARameter:NCAScading:FILE <filepath>

Related Commands
RFGSignal:SPARameter:MODE

Arguments
filepath::= <string> defines the path to the S-Parameter file.

Returns
A single <filepath> string.

Examples
RFGSIGNAL:SPARAMETER:NCASCADING:FILE "C:\temp\myFile.s12p" sets the filepath to "C:\temp\myFile.s12p" for use during compilation.

RFGSIGNAL:SPARAMETER:NCASCADING:FILE? might return "C:\temp\myOtherFile.s6p", indicating the current filepath.
RFGSignal:SPARameter:NCAScading:LAYout

This command sets or returns the 4 port S-Parameter Matrix Configuration, in Non-Cascading mode.

Conditions
- S-Parameter Mode must be set to Non-Cascading.
- Number of Ports must be set to 4.
- Requires an S-Parameters license.

Group
- S-Parameters

Syntax
- RFGSignal:SPARameter:NCAScading:LAYout {TYPical|ALTernate}
- RFGSignal:SPARameter:NCAScading:LAYout?

Related Commands
- RFGSignal:SPARameter:MODE

Arguments
- TYPical or ALTernate: selects the S-Parameter Matrix.

<table>
<thead>
<tr>
<th>S-Parameter Matrix Typical</th>
<th>S-Parameter Matrix Alternate</th>
</tr>
</thead>
<tbody>
<tr>
<td>SDD11 SDD12 SDC11 SDC12</td>
<td>SCC11 SCC12 SCD11 SCD12</td>
</tr>
<tr>
<td>SDD21 SDD22 SDC21 SDC22</td>
<td>SCC21 SCC22 SCD21 SCD22</td>
</tr>
<tr>
<td>SCD11 SCD12 SCC11 SCC12</td>
<td>SDC11 SDC12 SDD11 SDD12</td>
</tr>
<tr>
<td>SCD21 SCD22 SCC21 SCC22</td>
<td>SDC21 SCD22 SDD21 SDD22</td>
</tr>
</tbody>
</table>

Returns
- TYP
- ALT

Examples
- RFGSIGNAL:SPARAMETER:NCASCADING:LAYOUT TYPICAL sets the 4 port configuration's Layout to Typical, in Non-Cascading mode.
- RFGSIGNAL:SPARAMETER:NCASCADING:LAYOUT? might return TYP, indicating that configuration's Layout for port 4 is set to Typical, in Non-Cascading mode.
RFGSignal:SPARameter:NCAScading:RX[n]

This command sets or returns the S-Parameter port assignment of the channel’s specified receiver port number (Rx-Port) in Non-Cascading mode and Single-Ended Signalling Scheme (where applicable).

**Conditions**

S-Parameter Mode must be set to Non-Cascading.

S-Parameter Signalling Scheme must be set to Single-Ended.

Requires an S-Parameters license.

**Group**

S-Parameters

**Syntax**

RFGSignal:SPARameter:NCAScading:RX[n] <port number>

RFGSignal:SPARameter:NCAScading:RX[n]?

**Related Commands**

RFGSignal:SPARameter:MODE  
RFGSignal:SPARameter:CASCading:STYPe  
RFGSignal:SPARameter:NCAScading:TYPE  
RFGSignal:SPARameter:NCAScading:TX[n]

**Arguments**

[n] ::= <NR1>. A variable value to define the receiver port number (Rx-Port) of the channel.

The actual range is dependent on the Number of Ports (Type).

Type = 1, then n = no value
Type = 2, then n = {1}
Type = 4 then n = {1 – 2}
Type = 6 then n = {1 – 3}
Type = 8 then n = {1 – 4}
Type = 12 then n = {1 – 6}

If omitted, n is interpreted as 1.

<port number> ::= <NR1>. A variable value to define the S-Parameter Port assigned to the specified Rx-Port of the channel.

The actual range is dependent on the Number of Ports (Type).

Type = 1, then <port number> = no value
Type = 2, then <port number> = {1 – 2}
Type = 4 then <port number> = {1 – 4}
Type = 6 then <port number> = {1 – 6}
Type = 8 then <port number> = {1 – 8}
Type = 12 then <port number> = {1 – 12}
Commands in alphabetical order

**Returns**
A single \(<NR1>\) value.

**Examples**

RFGSIGNAL:SPARAMETER:NCASCADING:RX2 4 assigns S-Parameter port 4 to the channel’s receiver port 2, in the Single-Ended, Non-Cascading mode.

RFGSIGNAL:SPARAMETER:NCASCADING:RX4? might return 6, indicating that S-Parameter Port 6 is assigned to the channel’s receiver port 4, in the Single-Ended, Non-Cascading mode.
RFGSignal:SPARameter:NCAScading:SSCHeme

This command sets or returns the S-Parameter Signalling Scheme, in Non-Cascading mode. Signalling Scheme is only available when the Number of Ports is set to 4, 8, or 12.

Conditions
S-Parameter Mode must be set to Non-Cascading.
Requires an S-Parameters license.

Group
S-Parameters

Syntax
RFGSignal:SPARameter:NCAScading:SSCHeme
{SENDed|DIFFerential}
RFGSignal:SPARameter:NCAScading:SSCHeme?

Related Commands
RFGSignal:SPARameter:MODE

Arguments
SENDed – Single Ended Signal Scheme
DIFFerential – Differential Signal Scheme

Returns
SEND
DIFF

Examples
RFGSIGNAL:SPARAMETER:NCASCADING:SSCHHEME DIFF sets the Signalling Scheme to Differential, in Non-Cascading mode.
RFGSIGNAL:SPARAMETER:NCASCADING:SSCHHEME? might return SEND, indicating that the Signalling Scheme is set to Single Ended, in Non-Cascading mode.
**RFGSignal:SPARameter:NCAScading:STYPe**

This command sets or returns S-Parameter signal type (victim or aggressor), in Non-Cascading mode. The number of ports must be either 8 or 12.

**Conditions**
- S-Parameter Mode must be set to Non-Cascading.
- Number of Ports must be either 8 or 12.
- Requires an S-Parameters license.

**Group**
- S-Parameters

**Syntax**

```
RFGSignal:SPARameter:NCAScading:STYPe
{VICTim|AGGRessor|BOTH}
RFGSignal:SPARameter:NCAScading:STYPe?
```

**Related Commands**
- RFGSignal:SPARameter:MODE

**Arguments**
- VICTim – enables the victim signal type.
- AGGRessor – enables the aggressor signal type.
- BOTH – enables the victim and aggressor signal types.

**Returns**
- VICT
- AGGR
- BOTH

**Examples**

```
RFGSIGNAL:SPARAMETER:NCASCADING:STYPE BOTH sets the signal type to include both the Victim and Aggressor signal types, in Non-Cascading mode.

RFGSIGNAL:SPARAMETER:NCASCADING:STYPE? might return AGGR, indicating that the S-Parameter signal type is currently set to be Aggressor, in Non-Cascading mode.
```
RFGSignal:SPARameter:NCAScading:TX[n]

This command sets or returns the S-Parameter port assignment of the channel’s specified transmission port number (Tx-Port) in Non-Cascading mode and Single-Ended Signalling Scheme (where applicable).

Conditions

S-Parameter Mode must be set to Non-Cascading.

S-Parameter Signalling Scheme must be set to Single-Ended (where applicable).

Requires an S-Parameters license.

Group

S-Parameters

Syntax

RFGSignal:SPARameter:NCAScading:TX[n] <port number>
RFGSignal:SPARameter:NCAScading:TX[n]?

Related Commands

RFGSignal:SPARameter:MODE
RFGSignal:SPARameter:CASCading:STYPe
RFGSignal:SPARameter:NCAScading:TYPE
RFGSignal:SPARameter:NCAScading:RX[n]

Arguments

[n] ::= <NR1>. A variable value to define the transmission port number (Tx-Port) of the channel.

The actual range is dependent on the Number of Ports (Type).

Type = 1, then n = no value
Type = 2, then n = \{1\}
Type = 4 then n = \{1 – 2\}
Type = 6 then n = \{1 – 3\}
Type = 8 then n = \{1 – 4\}
Type = 12 then n = \{1 – 6\}

If omitted, n is interpreted as 1.

<port number> ::= <NR1>. A variable value to define the S-Parameter Port assigned to the specified Tx-Port of the channel.

The actual range is dependent on the Number of Ports (Type).

Type = 1, then <port number> = no value
Type = 2, then <port number> = \{1 – 2\}
Type = 4 then <port number> = \{1 – 4\}
Type = 6 then <port number> = \{1 – 6\}
Type = 8 then <port number> = \{1 – 8\}
Type = 12 then <port number> = \{1 – 12\}
Returns

A single <NR1> value.

Examples

RFGSIGNAL:SPARAMETER:NCASCADING:TX2 4 assigns S-Parameter port 4 to the channel’s transmission port 2, in the Single-Ended, Non-Cascading mode.

RFGSIGNAL:SPARAMETER:NCASCADING:TX4? might return 6, indicating that S-Parameter Port 6 is assigned to the channel’s transmission port 4, in the Single-Ended, Non-Cascading mode.
**RFGSignal:SPARameter:NCAScading:TYPE**

This command sets or returns the S-Parameter number of ports, in Non-Cascading mode.

**Conditions** Requires an S-Parameters license.

**Group** S-Parameters

**Syntax** `RFGSignal:SPARameter:NCAScading:TYPE {1|2|4|6|8|12}`

**Arguments** `{1|2|4|6|8|12}` – defines the number of S-Parameter ports.

**Returns** A single <NR1> value.

**Examples**

`RFGSIGNAL:SPARAMETER:NCASCADING:TYPE 12` sets the S-Parameter type to a 12-Port system for Non-Cascading mode.

`RFGSIGNAL:SPARAMETER:NCASCADING:TYPE?` might return 6, indicating that the S-Parameter type is a 6-Port system for Non-Cascading mode.
**WPLugin:ACTive**

This command sets or returns the active waveform plug-in.

To use the RF Generic Signal commands in this document, the active waveform plug-in must be set to RF Generic Signal.

**Group**  
Control

**Syntax**  
WPLugin:ACTive <plug-in_name>  
WPLugin:ACTive?

**Arguments**  
<plug-in_name>::=<string>  
A single string representing the waveform plug-in name.  
"RF Generic Signal" is the proper string to activate the RF Generic Signal plug-in.

**Returns**  
<plug-in_name>::=<string>

**Examples**  
WPLUGIN:ACTIVE "RF Generic Signal" sets the RF Generic Signal plug-in as the active plug-in.  
WPLUGIN:ACTIVE? might return "RF Generic Signal", indicating RF Generic Signal is currently the active waveform plug-in.
RFGSignal:SCModulation:TON

This command sets or returns the Sub Carrier Modulation state (enabled or disabled) for all carriers in the carrier table.

**Group**  
Sub Carrier Modulation

**Syntax**  
RFGSignal:SCModulation:TON {1|0|ON|OFF}  
RFGSignal:SCModulation:TON?

**Arguments**  
OFF or 0 disables Sub Carrier Modulation. OFF or 0 is the default value.  
ON or 1 enables Sub Carrier Modulation.

**Returns**  
A single <Boolean> value.

**Examples**  
RFGSIGNAL:SCMODULATION:TON 1 enables Sub Carrier Modulation for all carriers.  
RFGSIGNAL:SCMODULATION:TON? might return 0, indicating that Sub Carrier Modulation is disabled.
RFGSignal:SCModulation:TYPE

This command sets or returns the Sub Carrier Modulation type for the carriers in the carrier table.

**Group**  
Sub Carrier Modulation

**Syntax**  
RFGSignal:SCModulation:TYPE {AM|FM|PM}  
RFGSignal:SCModulation:TYPE?

**Arguments**  
AM, FM, and PM are the Modulation types.

**Returns**  
AM, FM, or PM indicating the analog modulation type.

**Examples**  
RFGSIGNAL:SCMODULATION:TYPE  
RFGSIGNAL:SCMODULATION:TYPE FM sets the Sub Carrier Modulation type to FM for the carriers in the carrier table.  
RFGSIGNAL:SCMODULATION:TYPE? might return PM, indicating that the Sub Carrier Modulation type is set to Phase Modulation for the carriers in the carrier table.
**RFGSignal:SCModulation:CFRequency**

This command sets or returns the Sub Carrier Modulation Carrier Frequency for the carriers in the carrier table.

**Group**  
Sub Carrier Modulation

**Syntax**  
RFGSignal:SCModulation:CFRequency <carrier_frequency>  
RFGSignal:SCModulation:CFRequency?

**Arguments**  
<carrier_frequency>::= <NR3> value.

**Returns**  
A single <NR3> value.

**Examples**  
RFGSIGNAL:SCMODULATION:CFREQUENCY 10E6 sets the Sub Carrier Modulation Carrier Frequency to 10 MHz for the carriers in the carrier table.  
RFGSIGNAL:SCMODULATION:CFREQUENCY? might return 1.0000000000E+6, indicating that the Sub Carrier Modulation Carrier Frequency is set to 1 MHz for the carriers in the carrier table.
**RFGSignal:SCModulation:FMModulation:FDEViation**

This command sets or returns the Frequency Deviation for the FM Sub Carrier Modulation for the carriers in the carrier table.

**Group**  
Sub Carrier Modulation

**Syntax**  
RFGSignal:SCModulation:FMModulation:FDEViation
<freq_deviation>
RFGSignal:SCModulation:FMModulation:FDEViation?

**Arguments**  
<freq_deviation>::= <NR3> value.

**Returns**  
A single <NR3> value.

**Examples**  
RFGSIGNAL:SCMODULATION:FMMODULATION:FDEVIATION 1E3 sets the FM Frequency Deviation to 1 kHz for the Sub Carrier Modulation for the carriers in the carrier table.

RFGSIGNAL:SCMODULATION:FMMODULATION:FDEVIATION? might return 1.4000000000E+3, indicating that the FM Frequency Deviation for the Sub Carrier Modulation is set to 1.4 kHz for the carriers in the carrier table.
RFGSignal:SCModulation:PMModulation:PDEViation

This command sets or returns the Phase Deviation for the PM Sub Carrier Modulation for the carriers in the carrier table.

Group Sub Carrier Modulation

Syntax RFGSignal:SCModulation:PMModulation:PDEViation <phase_deviation>

Related Commands RFGSignal:SCModulation:TYPE

Arguments <phase_deviation>::= <NR3> value.

Returns A single <NR2> value.

Examples RFGSIGNAL:SCMODULATION:PMMODULATION:PDEVATION 11.5 sets the PM Phase Deviation to 11.5° for the Sub Carrier Modulation for the carriers carrier table.

RFGSIGNAL:SCMODULATION:PMMODULATION:PDEVATION? might return 10.0000000000, indicating that the PM Phase Deviation for Sub Carrier Modulation is set to 10° for the carriers in the carrier table.
RFGSignal:SCModulation:AMModulation:AMIndex

This command sets or returns the AM Index for the AM Sub Carrier Modulation for the carriers in the carrier table.

Group Sub Carrier Modulation

Syntax

RFGSignal:SCModulation:AMModulation:AMIndex <AM_index>
RFGSignal:SCModulation:AMModulation:AMIndex?

Related Commands RFGSignal:SCModulation:TYPE

Arguments 

<AM_index>::= <NR2> value.

Returns A single <NR2> value.

Examples

RFGSIGNAL:SCMODULATION:AMMODULATION:AMINDEX 3.5 sets the AM Index to 3.5% for the Sub Carrier Modulation for the carriers carrier table.

RFGSIGNAL:SCMODULATION:AMMODULATION:AMINDEX? might return 10.0000000000, indicating that the AM Index for Sub Carrier Modulation is set to 10% for the carriers in the carrier table.
Index

C
command groups, 2-3

R
RFGSignal:CARRier:ADD, 2-17
RFGSignal:CARRier:COUNt?, 2-18
RFGSignal:CARRier[n]:AMODulation:AMModulation:AMNIndex, 2-18
RFGSignal:CARRier[n]:AMODulation:FMModulation:FDEViation, 2-19
RFGSignal:CARRier[n]:AMODulation:MFRequency, 2-20
RFGSignal:CARRier[n]:AMODulation:MSIGnal:FNAMe, 2-21
RFGSignal:CARRier[n]:AMODulation:MSIGnal:INTerpolation, 2-22
RFGSignal:CARRier[n]:AMODulation:MSIGnal:POFFset, 2-23
RFGSignal:CARRier[n]:AMODulation:MSIGnal:SRATe, 2-24
RFGSignal:CARRier[n]:AMODulation:MSIGnal:TYPe, 2-25
RFGSignal:CARRier[n]:AMODulation:PMModulation:PM3K, 2-26
RFGSignal:CARRier[n]:AMODulation:PMModulation:PM5K, 2-27
RFGSignal:CARRier[n]:AMODulation:PMModulation:TYPe, 2-28
RFGSignal:CARRier[n]:CMODulation:ADDMap, 2-29
RFGSignal:CARRier[n]:CMODulation:DELMap, 2-30
RFGSignal:CARRier[n]:CMODulation:ISYMbol, 2-31
RFGSignal:CARRier[n]:CMODulation:MMODE, 2-32
RFGSignal:CARRier[n]:CMODulation:MOFFset, 2-33
RFGSignal:CARRier[n]:CMODulation:QSYMbol, 2-34
RFGSignal:CARRier[n]:CMODulation:SELMap, 2-35
RFGSignal:CARRier[n]:CMODulation:SRATe, 2-36
RFGSignal:CARRier[n]:DATA, 2-37
RFGSignal:CARRier[n]:DATA:CODing, 2-38
RFGSignal:CARRier[n]:DATA:FILE, 2-39
RFGSignal:CARRier[n]:DATA:PRBS, 2-40
RFGSignal:CARRier[n]:DATA:PRBS:UDEFined:POLynominal, 2-42
RFGSignal:CARRier[n]:DATA:PRBS:UDEFined:RESet, 2-43
RFGSignal:CARRier[n]:DATA:PRBS:UDEFined:SREGister, 2-44
RFGSignal:CARRier[n]:DELETE, 2-45
RFGSignal:CARRier[n]:DISTortion:AMPLifier:AM3K, 2-46
RFGSignal:CARRier[n]:DISTortion:AMPLifier:AM5K, 2-47
RFGSignal:CARRier[n]:DISTortion:AMPLifier:LLEVel, 2-48
RFGSignal:CARRier[n]:DISTortion:AMPLifier:OPPOInt, 2-49
RFGSignal:CARRier[n]:DISTortion:AMPLifier:PM3K, 2-50
RFGSignal:CARRier[n]:DISTortion:AMPLifier:PM5K, 2-51
RFGSignal:CARRier[n]:DISTortion:AMPLifier:TON, 2-52
RFGSignal:CARRier[n]:DISTortion:AMPLifier:TYPe, 2-53
RFGSignal:CARRier[n]:DMODulation:APSK, 2-54
RFGSignal:CARRier[n]:DMODulation:APSK:PROTation, 2-55
RFGSignal:CARRier[n]:DMODulation:APSK:RADius, 2-56
RFGSignal:CARRier[n]:DMODulation:APSK:SRINg, 2-57
RFGSignal:CARRier[n]:DMODulation:APSK:SYMBoles, 2-58
RFGSignal:CARRier[n]:DMODulation:ASK:MINDex, 2-59
RFGSignal:CARRier[n]:DMODulation:CPM:MINDex, 2-60
RFGSignal:CARRier[n]:DMODulation:FSK, 2-61
RFGSignal:CARRier[n]:DMODulation:FSK:PDEViation, 2-62
RFGSignal:CARRier[n]:DMODulation:NDPSK:INDex, 2-63
Index

RFGSignal:Carrier[n]:DMODulation:NDPSK: PROTation, 2-64
RFGSignal:Carrier[n]:DMODulation:PSK, 2-65
RFGSignal:Carrier[n]:DMODulation:QAM, 2-66
RFGSignal:Carrier[n]:DMODulation:SRATe, 2-67
RFGSignal:Carrier[n]:DMODulation:TYPe, 2-68
RFGSignal:Carrier[n]:FILTer:ALPHa, 2-69
RFGSignal:Carrier[n]:FILTer:CLENgth, 2-70
RFGSignal:Carrier[n]:FILTer:FILE, 2-71
RFGSignal:Carrier[n]:FILTer:TYPe, 2-72
RFGSignal:Carrier[n]:FILTer:WINDow: CRIPple, 2-73
RFGSignal:Carrier[n]:FILTer:WINDow: KAISer, 2-74
RFGSignal:Carrier[n]:FILTer:WINDow: TYPe, 2-75
RFGSignal:Carrier[n]:FREQuency, 2-76
RFGSignal:Carrier[n]:HOPPing[:LIST]:ADD, 2-78
RFGSignal:Carrier[n]:HOPPing[:LIST]: AOFFset, 2-79
RFGSignal:Carrier[n]:HOPPing[:LIST]: COUNt?, 2-80
RFGSignal:Carrier[n]:HOPPing[:LIST]: DELete, 2-81
RFGSignal:Carrier[n]:HOPPing[:LIST]: FOFFset, 2-82
RFGSignal:Carrier[n]:HOPPing[:LIST]: FREQuency, 2-83
RFGSignal:Carrier[n]:HOPPing[:LIST]: HDURation, 2-84
RFGSignal:Carrier[n]:HOPPing[:LIST]: INSert, 2-85
RFGSignal:Carrier[n]:HOPPing[:LIST]: SELect, 2-86
RFGSignal:Carrier[n]:HOPPing[:LIST]: SSINdex, 2-87
RFGSignal:Carrier[n]:HOPPing:CUSTom: RLIST, 2-77
RFGSignal:Carrier[n]:HOPPing:PATTern, 2-88
RFGSignal:Carrier[n]:HOPPing:PRBS, 2-89
RFGSignal:Carrier[n]:HOPPing:PRBS:UDEFined: POLYnomial, 2-90
RFGSignal:Carrier[n]:HOPPing:PRBS:UDEFined: RESet, 2-91
RFGSignal:Carrier[n]:HOPPing:PRBS:UDEFined: SREGister, 2-92
RFGSignal:Carrier[n]:HOPPing:RANGe[:]: FREQuency:MAXimum, 2-94
RFGSignal:Carrier[n]:HOPPing:RANGe[:]: FREQuency:MINimum, 2-95
RFGSignal:Carrier[n]:HOPPing:RANGe[:]: FREQuency:SPACing, 2-96
RFGSignal:Carrier[n]:HOPPing:RANGe:ALISt[:]: ENABLE, 2-93
RFGSignal:Carrier[n]:HOPPing:TIME, 2-97
RFGSignal:Carrier[n]:HOPPing:TIME: HPSecond, 2-98
RFGSignal:Carrier[n]:HOPPing:TIME: SPHop, 2-99
RFGSignal:Carrier[n]:HOPPing:TON, 2-100
RFGSignal:Carrier[n]:INTerference:ANOIse: BANDwidth, 2-101
RFGSignal:Carrier[n]:INTerference:ANOIse: EBNo, 2-102
RFGSignal:Carrier[n]:INTerference:ANOIse: SNR, 2-103
RFGSignal:Carrier[n]:INTerference:ANOIse: TON, 2-104
RFGSignal:Carrier[n]:INTerference: FOFFset, 2-105
RFGSignal:Carrier[n]:INTerference:FOFFset: TON, 2-106
RFGSignal:Carrier[n]:INTerference:SINusoidal: CI, 2-107
RFGSignal:Carrier[n]:INTerference:SINusoidal: COFFset, 2-108
RFGSignal:Carrier[n]:INTerference:SINusoidal: TON, 2-109
RFGSignal:Carrier[n]:IQAMplitude, 2-110
RFGSignal:Carrier[n]:IQIMpairment:CLEAkage: IOFFset, 2-111
RFGSignal:Carrier[n]:IQIMpairment:CLEAkage: IQOFfset, 2-112
RFGSignal:Carrier[n]:IQIMpairment:CLEAkage: QOFFset, 2-113
RFGSignal:Carrier[n]:IQIMpairment:CLEAkage: TON, 2-114
RFGSignal:Carrier[n]:IQIMpairment:IQIMbalance: IMBalance, 2-115
RFGSignal:Carrier[n]:IQIMpairment:IQIMbalance: TON, 2-116
RFGSignal:Carrier[n]:IQIMpairment:IQSWap: TON, 2-117

Index-2 RF Generic Signals Plug-in Programmer Manual
Index

RFGSignal:CARRier[n]:IQIMpairment:NLDistortion:
  AM2K, 2-118
RFGSignal:CARRier[n]:IQIMpairment:NLDistortion:
  AM3K, 2-119
RFGSignal:CARRier[n]:IQIMpairment:NLDistortion:
  PM2K, 2-120
RFGSignal:CARRier[n]:IQIMpairment:NLDistortion:
  PM3K, 2-121
RFGSignal:CARRier[n]:IQIMpairment:QERRor:
  TON, 2-122
RFGSignal:CARRier[n]:IQIMpairment:QERRor:
  ERor, 2-123
RFGSignal:CARRier[n]:IQIMpairment:QERRor:
  TON, 2-124
RFGSignal:CARRier[n]:MULTipath:ADDPath, 2-125
RFGSignal:CARRier[n]:MULTipath:AMPLitude, 2-126
RFGSignal:CARRier[n]:MULTipath:DELay, 2-127
RFGSignal:CARRier[n]:MULTipath:DELPath, 2-128
RFGSignal:CARRier[n]:MULTipath:PHASe, 2-129
RFGSignal:CARRier[n]:MULTipath:SELPath, 2-130
RFGSignal:CARRier[n]:MULTipath:TURNon, 2-131
RFGSignal:CARRier[n]:NOISe:BANDwidth, 2-132
RFGSignal:Carrier[n]:PHASe, 2-133
RFGSignal:CARRier[n]:PRAMp:DTYPe, 2-134
RFGSignal:CARRier[n]:PRAMp:DURation, 2-135
RFGSignal:CARRier[n]:PRAMp:FUNCtion, 2-136
RFGSignal:CARRier[n]:PRAMp:ILEVel, 2-137
RFGSignal:CARRier[n]:PRAMp:LEVel:ADD, 2-138
RFGSignal:CARRier[n]:PRAMp:LEVel:DELETE, 2-139
RFGSignal:CARRier[n]:PRAMp:LEVel:DUrATION, 2-140
RFGSignal:CARRier[n]:PRAMp:LEVel:ESYMbol, 2-141
RFGSignal:CARRier[n]:PRAMp:LEVel:POWer, 2-142
RFGSignal:CARRier[n]:PRAMp:LEVel:SELect, 2-143
RFGSignal:CARRier[n]:PRAMp:LEVel:SSYMbol, 2-144
RFGSignal:CARRier[n]:PRAMp:PEXTend, 2-145
RFGSignal:CARRier[n]:PRAMp:TON, 2-146
RFGSignal:CARRier[n]:RFAMplitude, 2-147
RFGSignal:CARRier[n]:TYPE, 2-148
RFGSignal:COMPile, 2-149
RFGSignal:COMPile:AWARound, 2-150
RFGSignal:COMPile:CANCe, 2-151
RFGSignal:COMPile:CASSign, 2-152
RFGSignal:COMPile:CHANnel:I, 2-153
RFGSignal:COMPile:CHANnel:IQ, 2-154
RFGSignal:COMPile:CHANnel:Q, 2-155
RFGSignal:COMPile:CHANnel:RF, 2-156
RFGSignal:COMPile:CORRection:APPLY, 2-157
RFGSignal:COMPile:CORRection:PATH, 2-158
RFGSignal:COMPile:CORRection:PATH:I, 2-159
RFGSignal:COMPile:CORRection:PATH:IQ, 2-160
RFGSignal:COMPile:CORRection:PATH:Q, 2-161
RFGSignal:COMPile:CORRection:TYPE, 2-162
RFGSignal:COMPile:DUPConverter, 2-163
RFGSignal:COMPile:FDRange, 2-164
RFGSignal:COMPile:IChannel, 2-165
RFGSignal:COMPile:MARKer:ENABLE, 2-166
RFGSignal:COMPile:MARKer[n]:
  CFrequency, 2-167
RFGSignal:COMPile:MARKer[n]:MDATA, 2-168
RFGSignal:COMPile:NAME, 2-169
RFGSignal:COMPile:OSAMpling, 2-170
RFGSignal:COMPile:PLAY, 2-170
RFGSignal:COMPile:QChannel, 2-172
RFGSignal:COMPile:RFCchannel, 2-173
RFGSignal:COMPile:SFOMat, 2-174
RFGSignal:COMPile:SRATe, 2-175
RFGSignal:COMPile:SRATe:AUTO, 2-176
RFGSignal:COMPile:WLENgth, 2-177
RFGSignal:COMPile:WLENgth:AUTO, 2-178
RFGSignal:COMPile:WLENgth:TYPE, 2-179
RFGSignal:COMPile:WOVERwrite, 2-180
RFGSignal:RESet, 2-181
RFGSignal:SCModulation:AMModulation:
  AMINDEX, 2-236
RFGSignal:SCModulation:CFrequency, 2-233
RFGSignal:SCModulation:FMModulation:
  FDEVIation, 2-234
RFGSignal:SCModulation:PMModulation:
  PDEVIation, 2-235
RFGSignal:SCModulation:TONE, 2-231
RFGSignal:SCModulation:TYPE, 2-232
RFGSignal:SPARameter:BANDwidth, 2-185
RFGSignal:SPARameter:BANDwidth:AUTO, 2-186
RFGSignal:SPARameter:CAPSuling:AGGRessor[n]:
  AMPLITUDE, 2-188
RFGSignal:SPARameter:CAPSuling:AGGRessor[n]:
  CTALK, 2-189
Index-4 RF Generic Signals Plug-in Programmer Manual

RFGSignal:SPARameter:CASCading:AGGRessor[n]:
  DRATe, 2-190
RFGSignal:SPARameter:CASCading:AGGRessor[n]:
  SIGNal, 2-191
RFGSignal:SPARameter:CASCading:AGGRessor[n]:
  SIGNal:FILE, 2-192
RFGSignal:SPARameter:CASCading:AGGRessor[n]:
  SIGNal:PRBS, 2-193
RFGSignal:SPARameter:CASCading:AGGRessor2[:
  ENABle], 2-187
RFGSignal:SPARameter:CASCading:DEEMbed, 2-194
RFGSignal:SPARameter:CASCading:STAGE[m]:
  DRX[n], 2-195
RFGSignal:SPARameter:CASCading:STAGE[m]:
  DTX[n], 2-197
RFGSignal:SPARameter:CASCading:STAGE[m]:
  ENABle, 2-199
RFGSignal:SPARameter:CASCading:STAGE[m]:
  FILE, 2-200
RFGSignal:SPARameter:CASCading:STAGE[m]:
  RX[n], 2-201
RFGSignal:SPARameter:CASCading:STAGE[m]:
  SSCHeme, 2-203
RFGSignal:SPARameter:CASCading:STAGE[m]:
  TX[n], 2-204
RFGSignal:SPARameter:CASCading:STYPe, 2-206
RFGSignal:SPARameter:CASCading:TYPE, 2-207
RFGSignal:SPARameter:MODE, 2-208
RFGSignal:SPARameter:NCASCading:
  AGGRessor[n]:AMPLitude, 2-210
RFGSignal:SPARameter:NCASCading:
  AGGRessor[n]:CTALK, 2-211
RFGSignal:SPARameter:NCASCading:
  AGGRessor[n]:DRAFe, 2-212
RFGSignal:SPARameter:NCASCading:
  AGGRessor[n]:SIGNal, 2-213
RFGSignal:SPARameter:NCASCading:
  AGGRessor[n]:SIGNal:FILE, 2-214
RFGSignal:SPARameter:NCASCading:
  AGGRessor[n]:SIGNal:PRBS, 2-215
RFGSignal:SPARameter:NCASCading:AGGRessor2[:
  ENABle], 2-209
RFGSignal:SPARameter:NCASCading:
  DEEMbed, 2-216
RFGSignal:SPARameter:NCASCading:
  DRX[n], 2-217
RFGSignal:SPARameter:NCASCading:DTX[n], 2-219
RFGSignal:SPARameter:NCASCading:FILE, 2-221
RFGSignal:SPARameter:NCASCading:LAYout, 2-222
RFGSignal:SPARameter:NCASCading:RX[n], 2-223
RFGSignal:SPARameter:NCASCading:
  SSCHeme, 2-225
RFGSignal:SPARameter:NCASCading:STYPe, 2-226
RFGSignal:SPARameter:NCASCading:TYPE, 2-227
RFGSignal:SPARameter:SFORmat, 2-181
RFGSignal:SPARameter:SFORmat:LIQ, 2-183
RFGSignal:SPARameter:SFORmat:TON, 2-184
W
WPLugin:ACTive, 2-230